A Finite Calculus Formulation of the Level Set Equation

S.R. Idelsohn
E. Oñate
S.R. Ransau

Publication CIMNE Nº-239, September 2003

A Finite Calculus Formulation of the Level Set Equation

S.R. Idelsohn
E. Oñate
S.R. Ransau

Publication CIMNE Nº-239, September 2003

International Center for Numerical Methods in Engineering
Gran Capitán s/n, 08034 Barcelona, Spain

A finite calculus formulation of the level set equation

Sergio R. Idelsohn[1,2], Eugenio Oñate[2] and Samuel R. Ransau[3]

[1] *International Center for Computational Methods in Engineering (CIMEC)*
Universidad Nacional del Litoral and CONICET,
Santa Fe, Argentina

[2] *International Center for Numerical Methods in Engineering (CIMNE)*
Universitat Politécnica de Catalunya (UPC), Gran Capitán, s/n,
08034 Barcelona, Spain

[3] *Department of Marine Technology,*
Norwegian University of Science and Technology (NTNU)
7491 Trondheim, Norway

A finite calculus formulation of the level set equation is presented. Quadratic Galerkin finite elements are used for spatial discretisation. A unique stabilization parameter is computed. A time stabilization parameter allowing the use of the forward Euler scheme with Courant number larger than one is presented.

1 Introduction

Free surface flows are significantly more difficult to solve than infinite fluid problems. The extra difficulty in solving free surface flows comes from the coupling between the incompressible flow equations and the free surface equation. The latter expresses the constraint that fluid particles belonging to this boundary must remain there. However, the position of this boundary is unknown and has to be determined as a part of the solution. It is hence, of prime importance to devise accurate and efficient numerical methods able to handle such phenomena.

Many techniques have been developed for the prediction of the evolution of free boundaries and they can be mainly classified into two types : interface tracking methods and interface capturing methods [18].

For marine applications, the interface tracking type of methods (such as volume of fluid, height function, line segment, arbitrary Lagrangian-Eulerian) has been dominant the past years. Here, we use the level set method, an interface capturing method, in order to predict the evolution of the interface. In the level set method, the interface is embedded in the zero level set of a smooth function. The evolution of this zero level set is realised through the solution of a pure convection equation. Since this equation is purely convective, it is crucial to introduce some stabilization in the numerical method. This stabilization is necessary in order to eliminate the well-known oscillatory behaviour observed in the numerical solution when Galerkin finite elements (or centered finite differences/volumes) are used. These poor stability properties are due to the inherent negative numerical diffusion of these discretisation schemes. One could eliminate these non-physical oscillations by refining the grid. However, this technique can rapidly leads to prohibitively expensive simulations in terms of computer time. Therefore, several methods to avoid these non-physical oscillations have been devised through the years. We can, for example, name the Artificial Diffusion method, the Streamline-Upwind Petrov-Galerkin (SUPG), the Galerkin Least-Square (GLS), the Subgrid Scale and the Finite Calculus (FIC), among others. Here, we make use of the Finite Calculus (FIC) procedure [14, 15, 16], based on a new concept of flow balance laws over a finite size domain, to solve and consequently stabilize the level set equation.

Another important aspect of the level set method is the need of a reinitialisation step. The necessity of this step arises from the fact that the level set function is taken to be a distance function. This step is difficult to perform numerically since the equation solved for the reinitialisation is a very nonlinear equation, and it is consequently wishable to avoid performing this reinitialisation. For this purpose, we will here define the level set function not as a distance but as a smooth function with a rapid change in its gradient through the interface.

The paper is organised as follows: section 2 presents briefly a level set formulation of the free surface equation as well as the finite calculus formulation of the level set equation. Section 3 presents the numerical procedure used, and some numerical examples are shon in section 4. In section 5, the FIC method is used in combination with the forward Euler scheme. This results in an explicit scheme with Courant number larger than unity.

2 Mathematical formulation

2.1 Governing equations

Consider the motion of a rigid body in a viscous incompressible fluid incluiding a free surface. The equations governing this fluid flow are the incompresible Navier-Stokes equations and can be written as follows:

$$\nabla \cdot \mathbf{u} = 0 \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\,\mathbf{u} = \frac{1}{\rho}\nabla \cdot \sigma + \mathbf{f} \tag{2}$$

where $\boldsymbol{\sigma}$ is the stress tensor given by:

$$\sigma_{ij} = -p\delta_{ij} + 2\mu S_{ij} \tag{3}$$

p is the pressure, S the rate-of-strain tensor and μ the dynamic viscosity. Further, $\mathbf{u}$ is the fluid velocity, ρ the fluid density, and $\mathbf{f}$ an external force considered as known.
For free surface flows, there is usually one more equation which is the kinematic free surface boundary condition allowing to track the position of the fluid surface in time. We will here use the level set method instead.

2.2 Level set formulation

The level set technique was first introduced by Osher & Sethian [17]. A general description of the method can be found in [19].
This method is based on the solution of the following transport equation:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) = 0 \tag{4}$$

For incompressible flows, equation (4) is equivalent to (making use of the continuity equation (1)):

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0 \tag{5}$$

The unknown ϕ represents the signed distance between a point and the interface. The value of ϕ at a point $\mathbf{x}$ indicates if $\mathbf{x}$ is in fluid 1 or fluid 2 as follows:

$$\begin{gathered}\phi > 0 \rightarrow \text{fluid 1 (air for example)}\\ \phi = 0 \rightarrow \text{Interface (free surface for example)}\\ \phi < 0 \rightarrow \text{fluid 2 (water for example)}\end{gathered}$$

The level set technique is often classified as an interface-capturing scheme. This might be due to the fact that in the earliest level set algorithms, the zero level set of ϕ was never found explicitly [20]. However, it is probably better to classify the more recent level set methods [23] as "Eulerian" interface-tracking methods. There, interfaces are no longer captured but tracked since the zero level set of ϕ is found explicitly as a part of the numerical algorithm. The term "Eulerian" refers then to the fact that the interface is not represented as a set of connected moving points but as an embedded boundary (the zero level set of the function ϕ).
There exist different ways of solving equation (5) (or equivalently (4)). It is possible to use a two-phase flow formulation where both air and water flow are computed as in [20, 21], [1], [4] and [25]. Other authors [25] used a one-phase flow formulation. We here make use of the two-phase flow formulation.

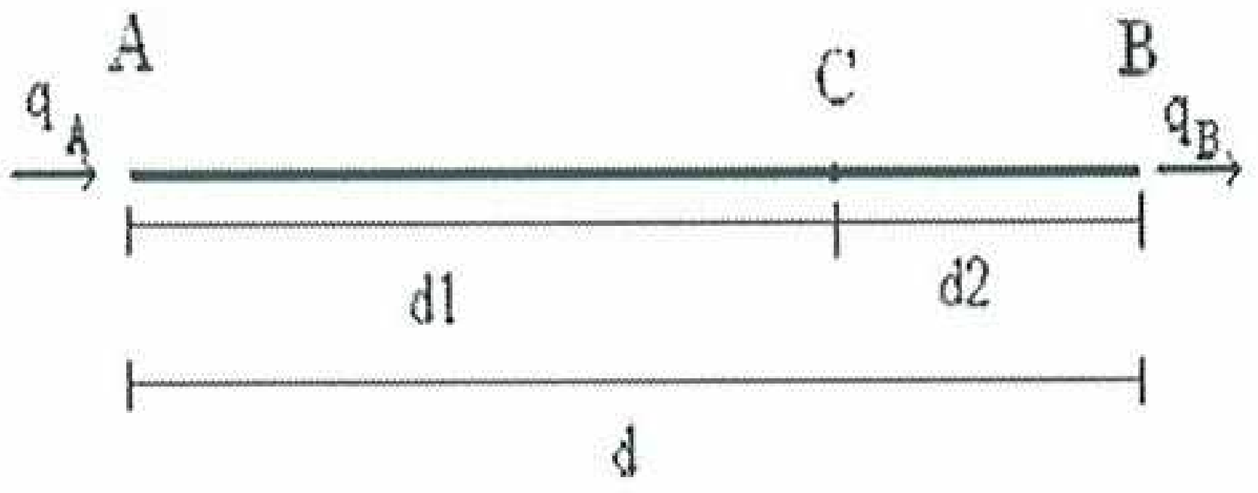


Figure 1: Balance of fluxes on a finite domain

2.3 Finite calculus formulation of the level set equation

The finite calculus method is based on flow balance over a domain of finite size. We will, in this section, briefly present the concept of the finite calculus method for a convection equation. For more details and for convection-diffusion equations, see [14, 15, 16].
Consider a steady pure convection problem in a 1D domain Ω of length L. The equation of balance of fluxes on a subdomain (of Ω) of length d (cf. figure 1) can be written as:

$$q_A - q_B = 0 \tag{6}$$

where q_A and q_B are the incoming and outgoing fluxes in A an B, respectively. In the case of a pure convection equation as the level set equation, the flux is: $q = -u\phi$ where ϕ is the transported quantity and u the transport velocity. (For convection-diffusion problems, the flux contains,in addition, a diffusive part of the form $\kappa \frac{\partial^2 u}{\partial x^2}$, where κ is the (positive) diffusivity coefficient).
We can, using Taylor series expansions, express the fluxes q_A and q_B in terms of the flux q in an arbitrary point C inside the subdomain of length d:

$$q_A = q_C - d_1 \frac{dq}{dx}|_C + d_1^2 \frac{d^2q}{dx^2}|_C + O(d_1^3) \tag{7}$$

and

$$q_B = q_C + d_2 \frac{dq}{dx}|_C + d_2^2 \frac{d^2q}{dx^2}|_C + O(d_2^3) \tag{8}$$

Inserting equations (7) and (8) into equation (6), leads to:

$$\frac{dq}{dx} - \frac{h}{2}\frac{d^2q}{dx^2} = 0 \tag{9}$$

where $h = d_1 - d_2$ and all derivatives are computed in point C. In standard calculus theory where the flux balance is written on an infinitively small domain, equation (9) reads to $\frac{dq}{dx} = 0$.

The term introduced by the FIC formulation (second term on the left-hand-side of equation (9)) can be interpreted as an artificial diffusion term, and consequently as a stabilizing term. This term is a consequence of the fact that the infinitesimal form of the balance equation is an unreacheable limit on a discrete grid [16].

The general (unsteady) finite calculus formulation of the level set equation is :

$$r - \frac{1}{2}\mathbf{h}^T \nabla r - \frac{1}{2}\lambda \frac{\partial r}{\partial t} = 0 \tag{10}$$

with

$$r = \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi \tag{11}$$

In equation (10), $\mathbf{h} = [h_x, h_y, h_z]^T$ (for a three-dimensional problem) contains the space stabilization parameters while λ contains the time stabilization. The vector $\mathbf{h}$ can be expressed in terms of the velocity vector $\mathbf{u}$ and $\nabla\phi$ as [15]:

$$\mathbf{h} = h_s \frac{\mathbf{u}}{|\mathbf{u}|} + h_t \frac{\nabla \phi}{|\nabla \phi|} \tag{12}$$

Further, h_s and h_t can be, for each element e, defined in function of a characteristic dimension $l^{(e)}$ of the element:

$$h_s^{(e)} = \alpha_s^{(e)} l^{(e)} \quad , \quad h_t^{(e)} = \alpha_t^{(e)} l^{(e)} \tag{13}$$

where $\alpha_s^{(e)}$ and $\alpha_t^{(e)}$ are the streamline and transverse stabilization parameter, respectively. Taking $\alpha_t^{(e)} = 0$ gives the same stabilization effects as in the SUPG method [2].

Any time integration (and in particular explicit schemes) can, in theory, be used with equation (10) since stability can be insured by choosing a suitable value for the parameter λ. We will come back to this in section 5.

A simpler version of the stabilized equation (10) can be found by retaining only the second order terms in space and time:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi - \frac{1}{2}\mathbf{h}^T \mathbf{u}\, \Delta \phi - \frac{1}{2}\lambda \frac{\partial^2 \phi}{\partial t^2} = 0 \tag{14}$$

An even simpler form can be found by only retaining the second order terms in space:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi - \frac{1}{2}\mathbf{h}^T \mathbf{u}\, \Delta \phi = 0 \tag{15}$$

Equation (15), is used for the examples shown in section 4, where the last terms on the left-hand-side of equation (15) is the space stabilization term. Stable time integration schemes then need to be used in order to solve equation (15) numerically.

Since the level set equation is a pure advection equation, the transverse stabilization parameter α_t can be set equal to zero in equation (15).

2.4 Stabilization parameters for two-dimensional quadratic elements

In [3], Codina showed that when quadratic elements are used in steady convection-diffusion problems, the number of optimal streamline stabilization parameters is 2: one for the extreme nodes of the elements and another one for the central nodes. He also showed that it was however possible to find a unique (non-optimal) stabilization parameter for both extreme and central nodes, given by:

$$\alpha_s(\gamma) = \frac{1}{2}\left(\coth\gamma - \frac{1}{\gamma}\right) \tag{16}$$

where γ is the Péclet number defined as $\gamma = \dfrac{u\Delta x}{2\kappa}$ and $\kappa > 0$ is the diffusion constant. In the case of a pure steady convection equation, $\gamma \rightarrow \infty$ and the limit of equation (16) as $\gamma \rightarrow \infty$ is $\dfrac{1}{2}$.

For equation (15) which is time-dependent, a possibility is to choose the same value of α_s as for the stationary problem. However numerical experiments ([3], [24]) have shown that this value is not optimal for time-dependent pure advection equation. Tezduyar & Ganjoo [24] proposed to define the stabilization parameter as:

$$\alpha_s = \frac{2}{\sqrt{15}} + \left(1 - \frac{2}{\sqrt{15}}\right)\frac{|\mathbf{u}|\Delta t}{\Delta x} \tag{17}$$

for one-dimensional pure convection equations when linear elements are used.

We use both equations (16) and (17) as well as numerical experiments performed (cf. section 4) to guide us in the choice of α_s, and take:

$$\alpha_s = \alpha\left[\frac{2}{\sqrt{15}} + \left(1 - \frac{2}{\sqrt{15}}\right)\frac{|\mathbf{u}|\Delta t}{\Delta x}\right] \tag{18}$$

with $\alpha = \dfrac{1}{2}$.

3 Numerical procedure

3.1 Time discretisation

In order to solve equation (15) numerically, we start by discretising it in time using a θ-rule, where an equation of the type:

$$\frac{\partial\phi}{\partial t} = G \tag{19}$$

is discretised by:

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \theta G^{n+1} + (1-\theta) G^n, \qquad 0 \leq \theta \leq 1 \tag{20}$$

In the case of equation (15), the right-hand-side G is given by:

$$-\mathbf{u} \cdot \nabla \phi + \frac{1}{2} \mathbf{h}^T \mathbf{u}\, \Delta \phi \tag{21}$$

Equation (20) is then:

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \theta \left[-\mathbf{u}^{n+1} \cdot \nabla \phi^{n+1} + \frac{1}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1}\, \Delta \phi^{n+1} \right] +$$

$$(1-\theta) \left[-\mathbf{u}^{n+1} \cdot \nabla \phi^n + \frac{1}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1}\, \Delta \phi^n \right], \qquad 0 \leq \theta \leq 1 \tag{22}$$

It is well-known that the scheme defined by equation (20) is first order accurate in time for any $\theta \in [0,1]$, except for $\theta = \frac{1}{2}$ where second order is obtained. It is interesting to note that for $\frac{1}{2} \leq \theta \leq 1$, the scheme given by equation (20), or in our case by equation (22) is unconditonally stable (and hence convergent). For $0 \leq \theta \leq \frac{1}{2}$ conditions on the time step exist in order to insure stability. Three cases are particularly interesting: $\theta = 0$, the forward Euler scheme; $\theta = 1$, backward Euler scheme; and $\theta = \frac{1}{2}$ the Crank-Nicholson scheme. The latter case is interesting because as already mentionned, second order time accuracy is obtained. $\theta = 1$ is interesting because it is unconditionally stable and possess important numerical damping. This can be useful to smooth oscillations appearing in problems where the solution develops sharp gradients. The case $\theta = 0$ is interesting since it gives rise to an explicit scheme and no solver for algebraic systems is needed. Codina [3] performed a stability analysis of the forward Euler scheme for convection-diffusion equation and showed that for quadratic elements, the advective limit of the stability condition is $c \leq \frac{1}{8}$ where $c = \frac{|\mathbf{u}| \Delta t}{l}$ is the Courant number.

3.2 Spatial discretisation

The space discretisation is performed by the standard Galerkin finite element method: we seek for a solution of the form

$$\phi(\mathbf{x}, t) = \sum_j \phi_j N_j(\mathbf{x}, t) \tag{23}$$

where ϕ_j are the nodal values and $N_j(\mathbf{x},t)$ are the shape functions.
The Galerkin formulation of equation (22) is:

$$\sum_j \int_\Omega \phi_j^{n+1} N_j N_i \, d\Omega - \int_\Omega \phi^n N_i \, d\Omega =$$
$$-\Delta t\theta \sum_j \int_\Omega \left[\mathbf{u}^{n+1} \cdot \nabla N_j \, N_i + \frac{1}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1} \, \nabla N_j \cdot \nabla N_i \right] \phi_j^{n+1} \, d\Omega \tag{24}$$
$$-\Delta t(1-\theta) \int_\Omega \left[\mathbf{u}^{n+1} \cdot \nabla \phi^n \, N_i + \frac{1}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1} \, \nabla \phi^n \cdot \nabla N_i \right] d\Omega$$

The condition that $\dfrac{\partial \phi}{\partial \mathbf{n}} = 0$ on the domain boundary has been used in the derivation of equation (24).
Equation (24) can be further written as:

$$\sum_j \int_\Omega \left[N_i N_j - \Delta t\theta \mathbf{u}^{n+1} \cdot \nabla N_j \, N_i - \frac{\Delta t\theta}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1} \, \nabla N_j \cdot \nabla N_i \right] \phi_j^{n+1} \, d\Omega =$$
$$\int_\Omega \left[N_i \phi^n - \Delta t(1-\theta) \mathbf{u}^{n+1} \cdot \nabla \phi^n \, N_i - \frac{\Delta t(1-\theta)}{2} (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1} \, \nabla \phi^n \cdot \nabla N_i \right] d\Omega \tag{25}$$

The right-hand-side of equation (25) can be put in a vector form ($\mathbf{b}_i^n$), while the left-hand-side can be put in the following matrix form:

$$\sum_j (M_{ij} + C_{ij} + S_{ij}) \, \phi_j^{n+1} \tag{26}$$

with

$$M_{ij} = \int_\Omega N_i N_j d\Omega \tag{27}$$

$$C_{ij} = -\Delta t \, \theta \int_\Omega \mathbf{u}^{n+1} \cdot \nabla N_j N_i d\Omega \tag{28}$$

$$S_{ij} = -\frac{\Delta t \, \theta}{2} \int_\Omega (\mathbf{h}^T)^{n+1} \mathbf{u}^{n+1} \, \nabla N_j \cdot \nabla N_i d\Omega \tag{29}$$

The stabilization parameters are defined in equations (12) and (13). We use $l^{(e)} = (\Omega^e)^{1/d}$ where d is the space dimension. Other expressions for the characteristic length can also be used [16]. Several values of the parameters α_s will be considered in the numerical examples.

3.3 Boundary conditions for the level set function

Even though in theory, the value of level set function ϕ should not be set on any boundary, in the following examples, it has been set in all nodes belonging to a boundary and where flow was flowing into the domain. This must be done in order to get a well-posed problem since equation (5) is a first order hyperbolic equation.

4 Numerical examples

We consider simple cases where the velocity field is given by an analytical function so that we can easily evaluate the accuracy and the effects of the FIC method.

4.1 Tests with an uniform velocity

First we consider the advection of a circle of radius r_0 under the velocity field given by:

$$\mathbf{u} = \mathbf{i} \tag{30}$$

where $\mathbf{i}$ is the unit vector in x-direction.
We look at the conservation of the circle. The latter should be advected without changing neither form nor radius. We will evaluate the change in volume defined by:

$$\Delta V = \frac{V(t) - V_0}{V_0} \tag{31}$$

where the volume at time t is defined by:

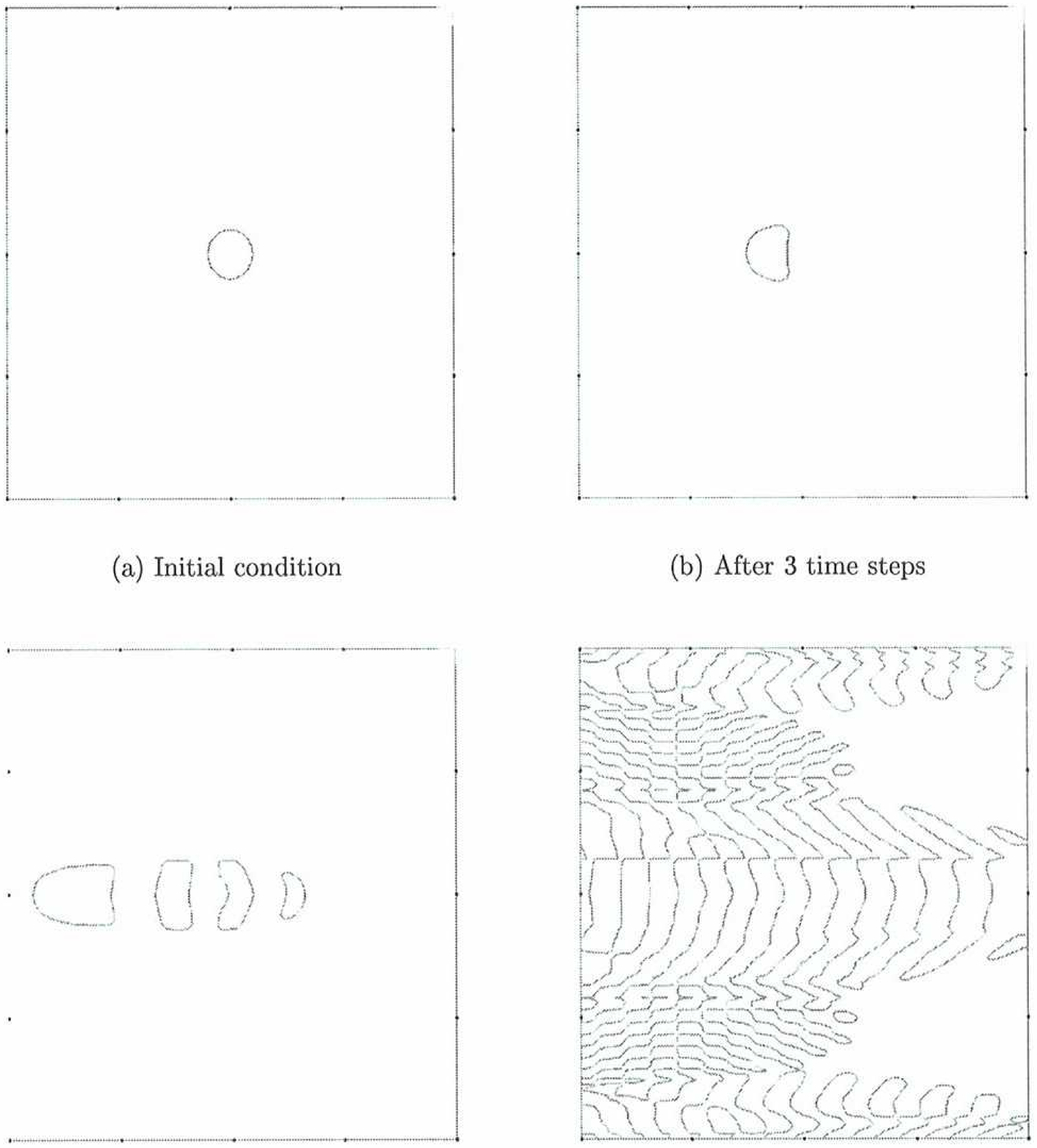
$$V(t) = \int_\Omega H(\phi)\, d\Omega\ , \tag{32}$$

V_0 is the initial volume, Ω the total fluid domain and H is the Heavyside function:

$$H(\phi) = \left\{ \begin{array}{ll} 1 & \text{if } \phi < 0 \\ 0 & \text{if } \phi > 0 \end{array} \right. \tag{33}$$

In the computations, we vary the parameter α (see equation (18)) between 0 to 1. For the time integration, we use $\theta = 1$ and $\theta = \dfrac{1}{2}$ in equation (22) denoted BE and CN, respectively.

Figure 2(a) shows the zero level set at the initial time. This initial condition will be used in all the two-dimensional examples presented below. Figures 2(b), 2(c) and 2(d) show the solution after 3, 10 and 50 time steps, respectively, computed with CN, when no stabilization is used ($\alpha = 0$). As expected, oscillations appear relatively fast.


(a) Initial condition

(b) After 3 time steps

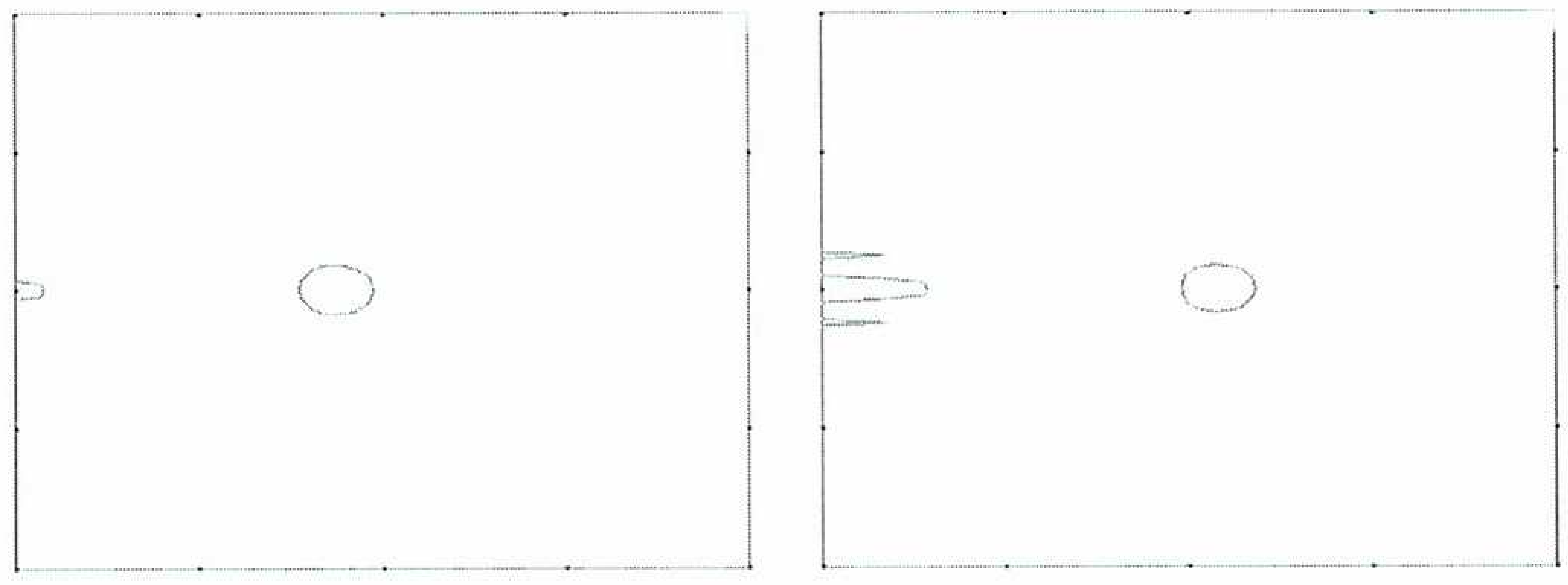
(c) After 10 time steps

(d) After 50 time steps

Figure 2: $\mathbf{u} = \mathbf{i}$. Computed solutions for $\alpha_s = 0$. CN. $\Delta t = 0.005$, $\Delta x = 0.02$

Figure 3 shows the effect of not prescribing a boundary on the level set field at boundaries where fluid is flowing in (the inlet boundary in the present example). Oscillations appear at this boundary.

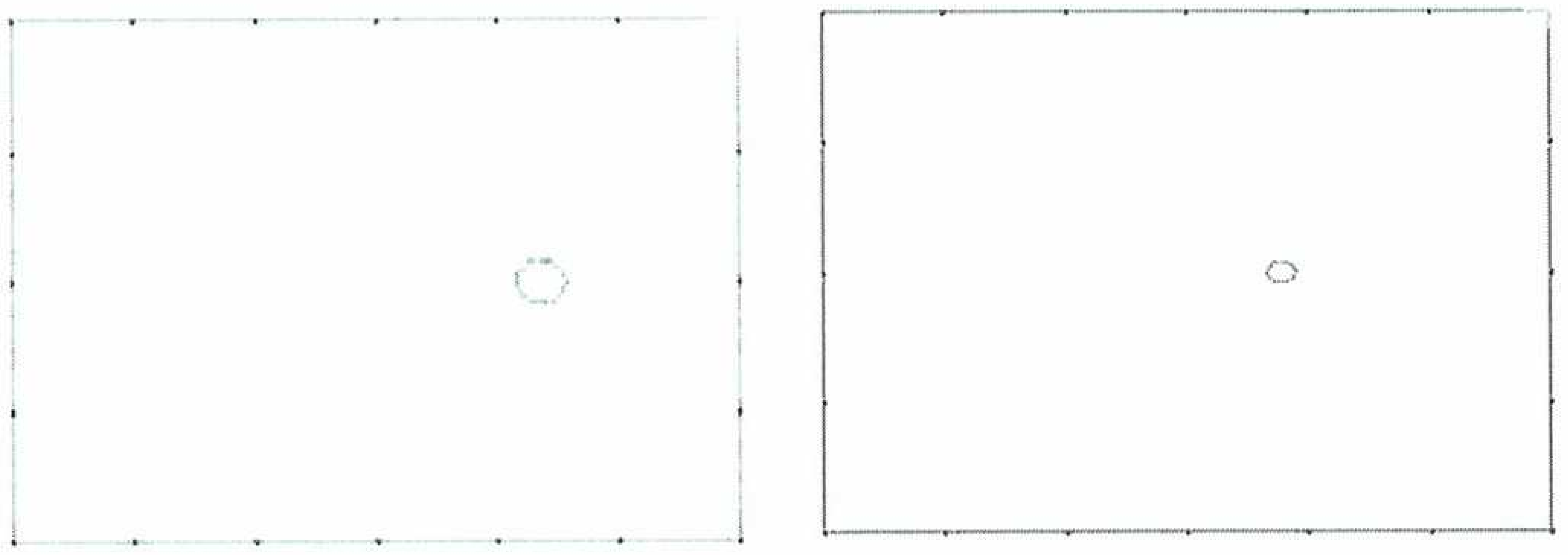
Figures 4 and 5 show that the value $\alpha = 1$ (cf. equation (18)) is over-diffusive for both CN and BE schemes. Figure 4 shows a relatively large change in the shape of the circle,



(a) After 35 time steps

(b) After 55 time steps

Figure 3: $\mathbf{u} = \mathbf{i}$. Computed solutions for $\alpha_s = 1.$ without inlet boundary on ϕ.



(a) Crank-Nicholson after 1.8s

(b) Backward Euler after 1.4s

Figure 4: $\mathbf{u} = \mathbf{i}$. Computed solutions for $\alpha_s = 1.$ $\Delta t = 0.005$, $\Delta x = 0.02$

as well as a change in volume. The latter was expected due to numerical diffusion. The former can be explained by the fact that quadratic elements were used, and as mentionned by Codina in [3], there exist in that case two optimal stabilization parameters in the streamline direction: one for the nodes inside the elements and one for the nodes at the boudary of the elements. This implies a non-constant added diffusion over the domain. However, only one stabilization parameter was used in the present implementation. This implies a larger distorsion of the flow quantities at the extreme nodes than in the case where the two optimal parammeters are used. The change in raduis is primarily in the

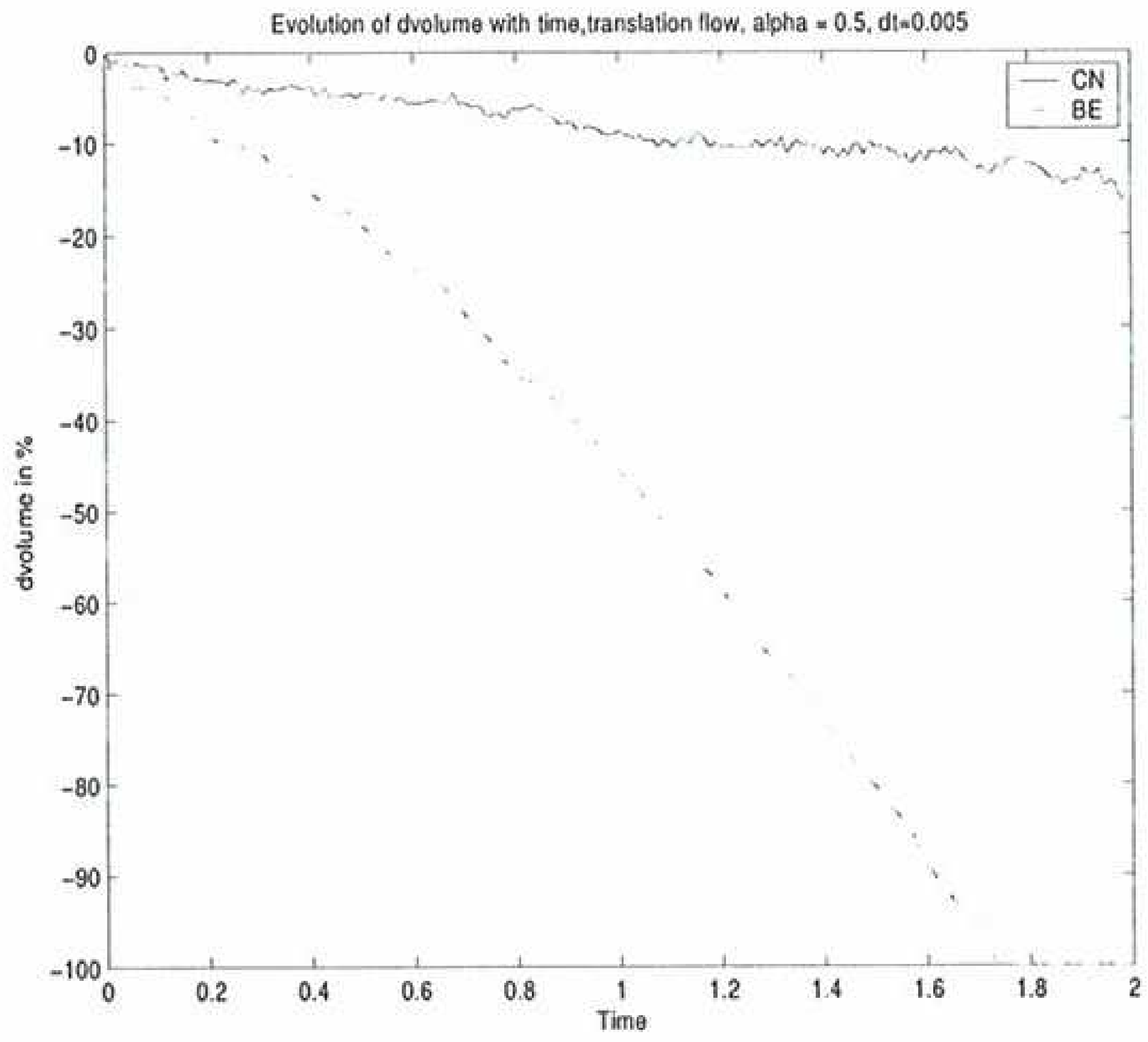
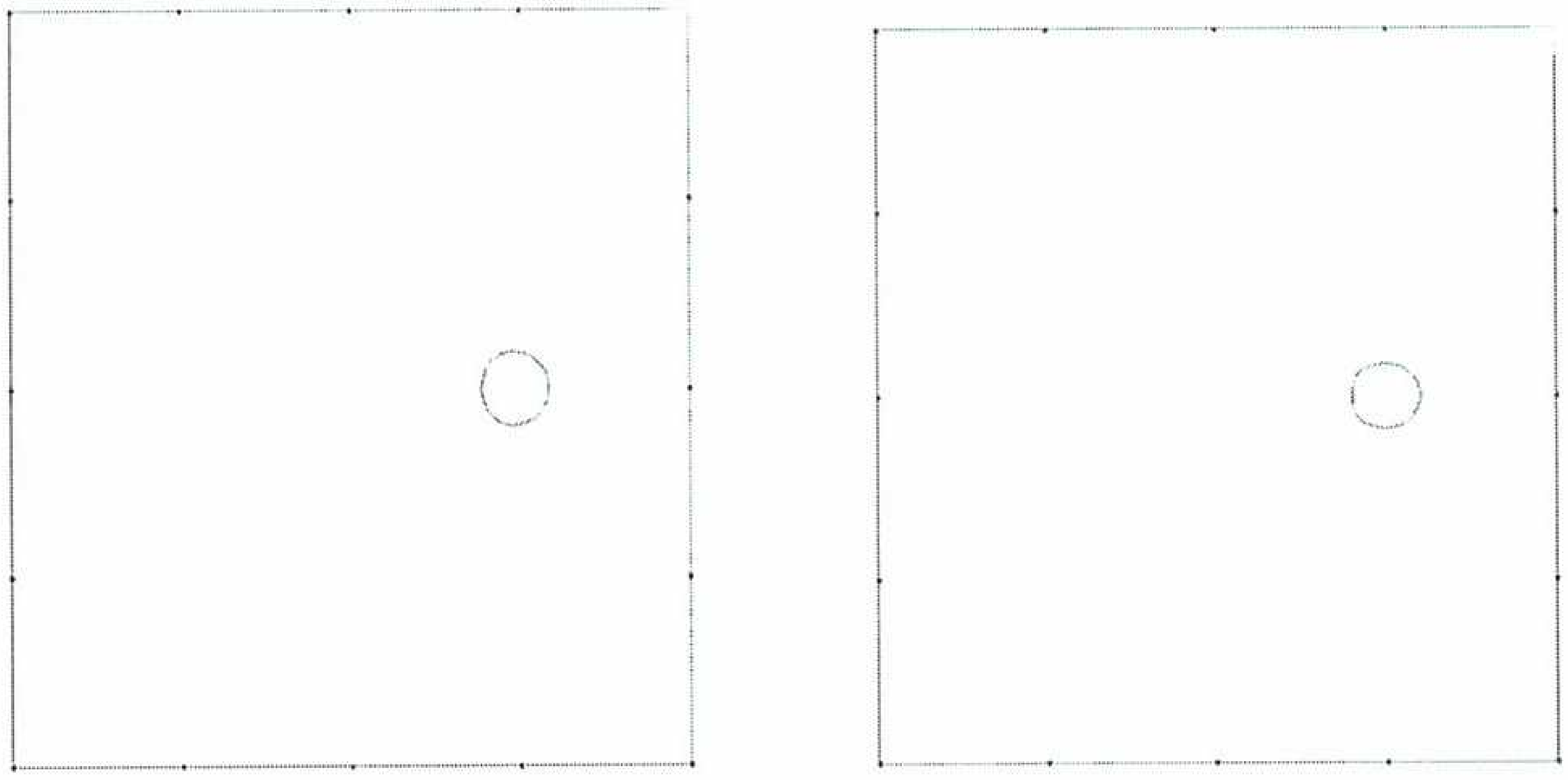


Figure 5: $\mathbf{u} = \mathbf{i}$. Evolution of ΔV for $\alpha_s = 1$. $\Delta t = 0.005$, $\Delta x = 0.02$



(a) Crank-Nicholson

(b) Backward Euler

Figure 6: $\mathbf{u} = \mathbf{i}$. Computed solutions after 0.5s for $\alpha_s = \dfrac{1}{2}$. $\Delta t = 0.005$, $\Delta x = 0.02$

direction of the flow which is normal since in the examples shown here, only the streamline stabilization parameter is different from zero.

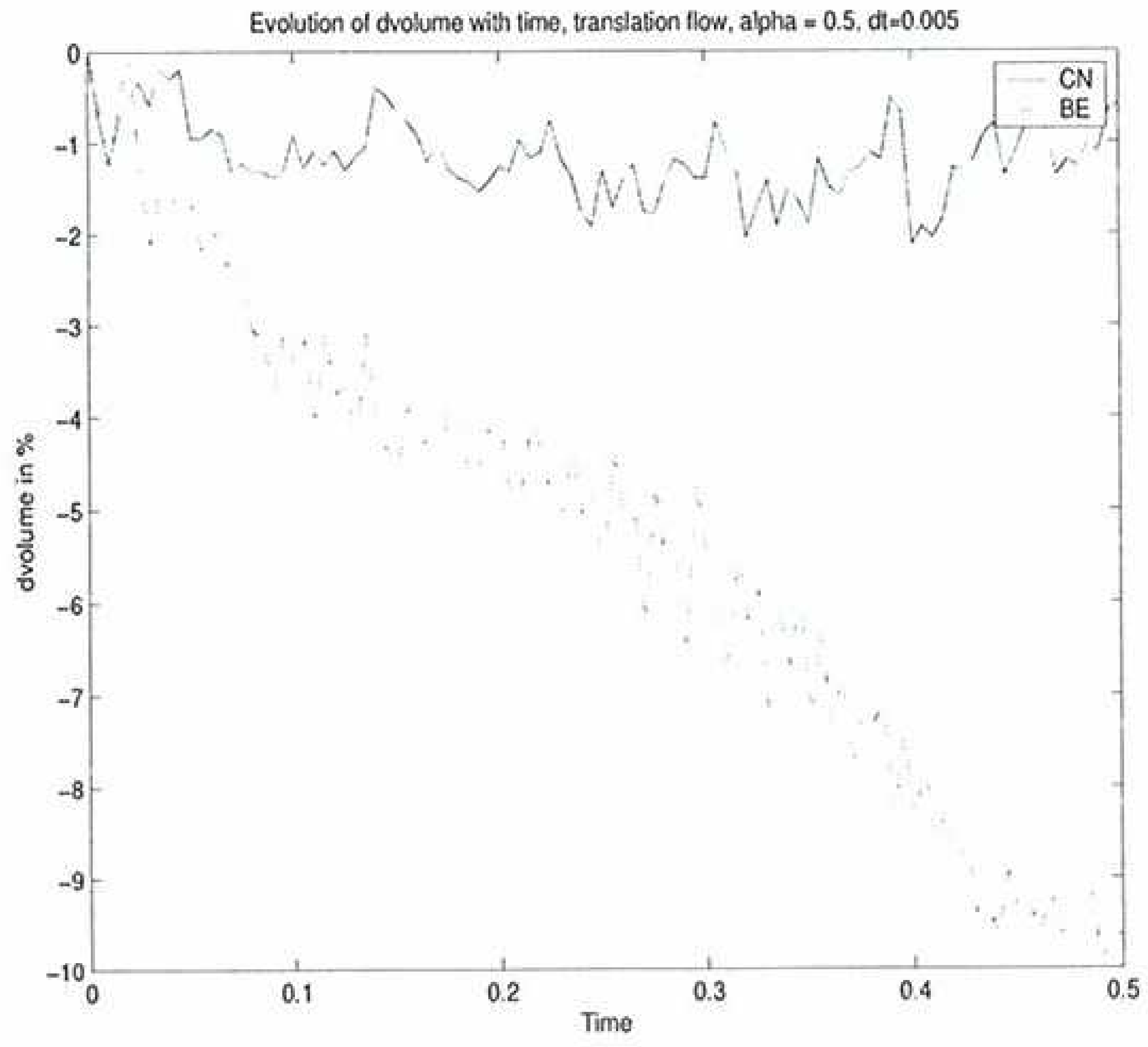


Figure 7: $\mathbf{u} = \mathbf{i}$. Evolution of ΔV for $\alpha_s = \frac{1}{2}$. $\Delta t = 0.005$, $\Delta x = 0.02$

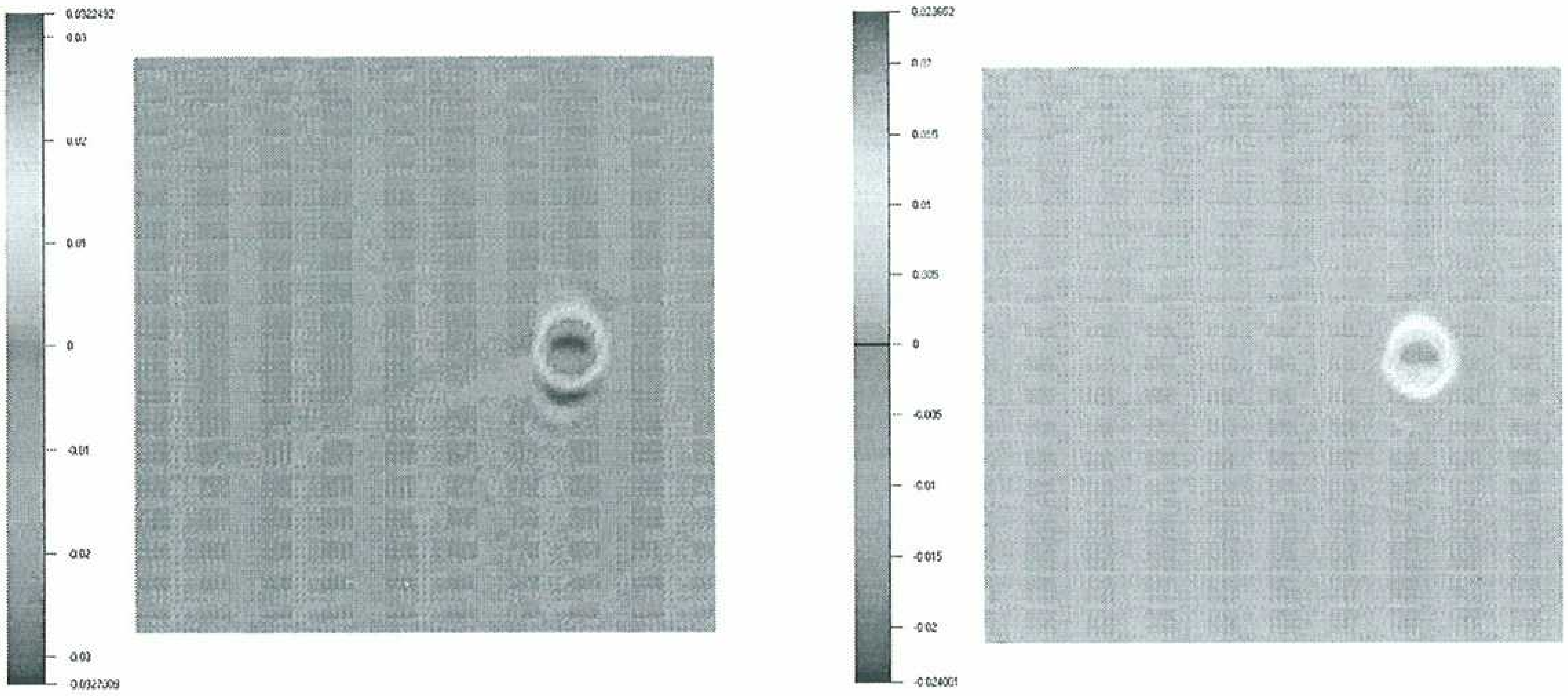
Figures 6 and 7 show the results for $\alpha = \frac{1}{2}$. Here, the results are relatively good, and the circle shape seems to be better conserved in that case. The mass loss is satisfactory in the case of the CN scheme. For the BE scheme, the mass loss is very important (until complete vanishment of the circle), but this first order implicit scheme is well-known for its strong diffusive behaviour.

4.2 Tests with a circular velocity

This test case is similar to the previous one, but this time the velocity field is given by:

$$\mathbf{u} = \pi r \mathbf{i}_\theta \tag{34}$$

where $\mathbf{i}_\theta = -\sin\theta\,\mathbf{i} + \cos\theta\,\mathbf{j}$.
For this case, we used different values for the stabilization parameter α: $\frac{1}{2}$, $\frac{1}{12}$ and 0. As expected, for the case $\alpha = 0$, strong oscillations appear rapidly. The value $\alpha = \frac{1}{2}$ produces reasonably good results with a mass loss of approximatively 2.5% after one period. With the value $\alpha = \frac{1}{12}$, the mass loss is slightly less (1.7% after one period) but the solution is less stable. After one period, the apparition of oscillations can be noted behind the circle (cf. figures 8(a)) in the simulation where $\alpha = \frac{1}{12}$ is used.
Comparing figures 8(a) and 8(b) on one hand, and figures 10(a) and 10(b) on the other hand, it can be noted that the smaller is α, the better is the volume conserved but the



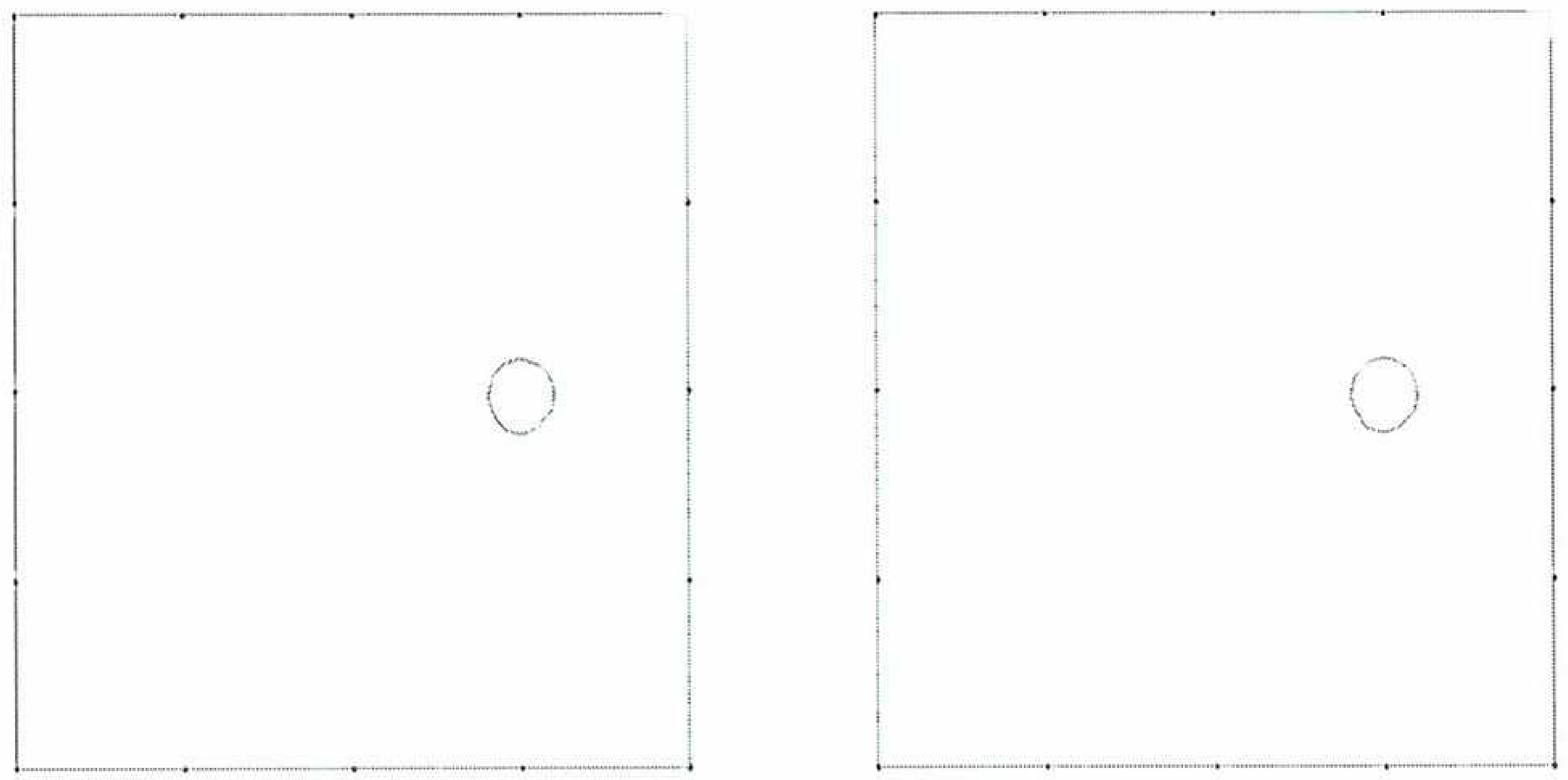
(a) $\alpha_s = \dfrac{1}{12}$

(b) $\alpha_s = \dfrac{1}{2}$

Figure 8: $\mathbf{u} = \pi r \mathbf{i}_\theta$. Computed level set field after one period CN scheme. $\Delta t = 0.005$, $\Delta x = 0.02$

less stable is the solution (results which was expected).

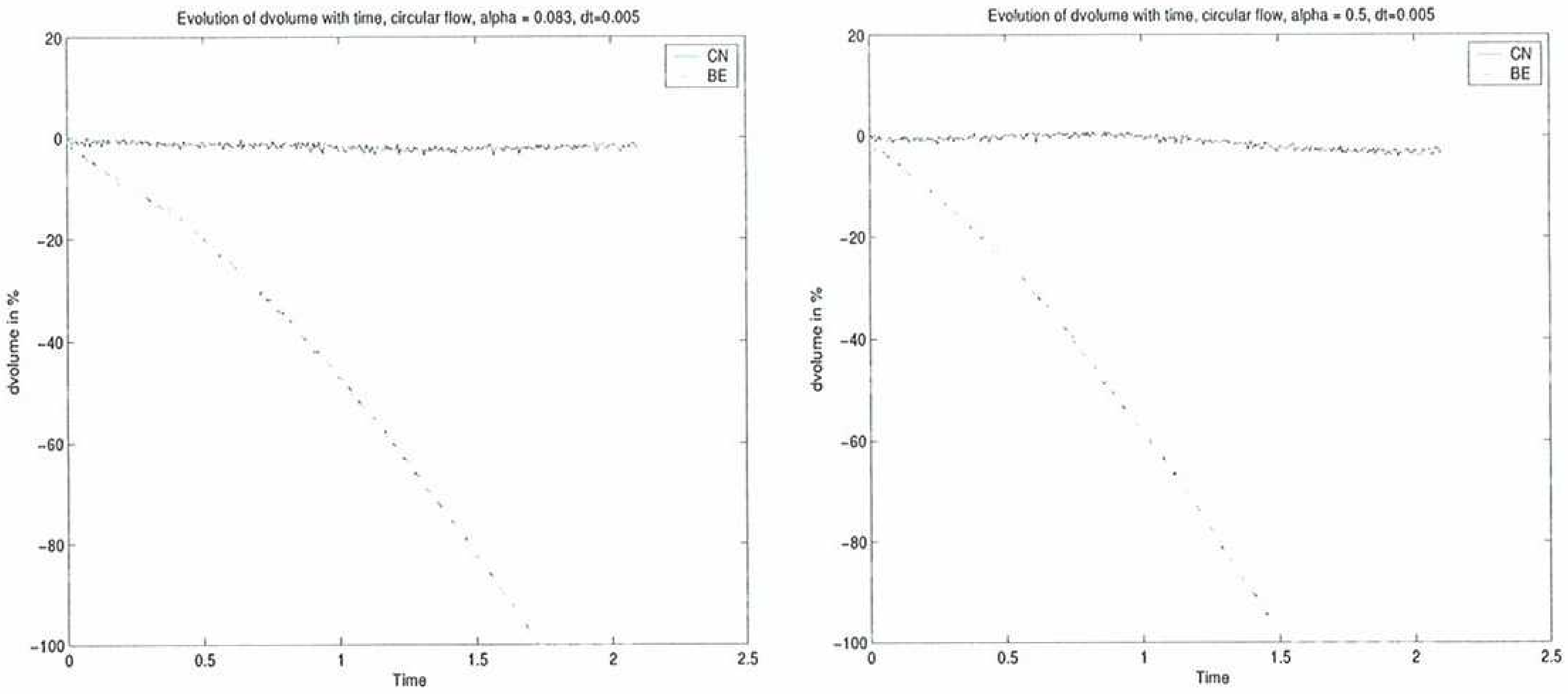
The value $\alpha = \dfrac{1}{12}$ cannot be used for long simulations as oscillations will develop and pollute the results. The relatively "late" apparition of oscillations with such a low value of the paameter α, might be explained by the fact that for the circular flow, the zero level of ϕ (that we are tracking) do not approach the boundary close enough such that the inherent unstable behaviour of the Galerkin method is trigged.
The value $\alpha = \dfrac{1}{2}$ seems therefore to be a better choice. The mass loss (2.5% in that case) is acceptable. If one is interested in better mass conservation properties, one can still reduce the value of alpha. However, one should not forget that the solution will perhaps oscillate after a certain number of time steps. One should also keep in mind that every stable numerical scheme contains numerical diffusion which will influence the mass conservation.



(a) $\alpha_s = \frac{1}{12}$

(b) $\alpha_s = \frac{1}{2}$

Figure 9: $\mathbf{u} = \pi r \mathbf{i}_\theta$. Computed zero level set after one period CN scheme. $\Delta t = 0.005$, $\Delta x = 0.02$



(a) $\alpha_s = \frac{1}{12}$

(b) $\alpha_s = \frac{1}{2}$

Figure 10: $\mathbf{u} = \pi r \mathbf{i}_\theta$. Evolution of ΔV. CN and BE schemes. $\Delta t = 0.005$, $\Delta x = 0.02$

5 Solution of the stabilized equation using the forward Euler scheme

As mentionned in section 3.1, using an explicit time integration scheme (as forward Euler) to solve equation (15) can be very attractive. However, this approach would lead to prohibitively small time steps for stability to hold. In order to be able to increase the time step, the general FIC formulation given by equation (10) can be used.
We will, in this section, study the stability of the difference equation resulting from the simplest second-order centred spatial differencing and first order forward temporal differencing (Euler), often referred as FTCS (forward time, centred space).

5.1 Problem statement and general consideration

Historically, two different concepts of stability have been used in order to study numerical solutions of ordinary differential equation: the matrix method and the von Neumann method.
In the matrix method, the differential equations are put in the form $x^{n+1} = Ex^n$ where E denotes an MxM matrix, x^n an M-dimensional vector and the superscript n denotes the time level. Since the error ϵ will also satisfy the difference equation (see [10]), one will have: $\epsilon^{n+1} = E\epsilon^n$. For the scheme to be stable, $||\epsilon^n||$ must be bounded for any n. Since

$$||\epsilon^{n+1}|| = ||E\epsilon^n|| = ||E^{n+1}\epsilon^0|| \leq ||E||^{n+1}\ ||\epsilon^0|| \tag{35}$$

stability will hold if:

$$||E|| \leq 1 \tag{36}$$

In fact only

$$||E|| \leq 1 + O(\Delta t) \tag{37}$$

is necessary [13], where Δt is the time step.
Condition (37) (or (36)) is usually impossible to check for any Δt, Δx and boundary condition. Therefore, instead of (37) one usually checks

$$\rho(E) \leq 1 \tag{38}$$

since one knows that $\rho(E) \leq ||E||$ for any matrix-norm.
This method was proven [6] to give stability condition less accurate than than the von Neumann analysis. We will therefore, in the sequel use the von Neumann method as Hindmarsh, Gresho and Griffiths [6].

In the von Neumann method, the error is expanded in Fourier series. The core of the method consists in analyzing these Fourier modes. This analysis requires periodicity in the space domain. When this periodicity does not hold, the von Neumann analysis only

gives necessary conditions for stability.
In the following, we will focus on the 1D initial-boundary-value problem:

Find a scalar function $\phi = \phi(x,t)$ in $[0,L] \times I\!R_+$ satisfying:

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} - K\frac{\partial^2 \phi}{\partial x^2} = 0 \tag{39}$$

with initial condition:

$$\phi(x,0) = \phi^0(x), \ \forall x \in]0,L[\tag{40}$$

and boundary conditions

$$\phi(0,t) = \phi(L,t) \quad \text{and} \quad \frac{\partial \phi}{\partial t}(0,t) = \frac{\partial \phi}{\partial t}(L,t), \ \ \forall t \in I\!R_+ \tag{41}$$

where $\phi^0(x)$ is a given function. Equation (41) comes from the fact that, as mentionned previously, periodicity in the space domain is required in the von Neumann analysis. The coefficient K is, in the case of a pure convection, given by $\frac{\alpha ul}{2}$ (see section 2.3), where l is the grid spacing. In the case of the of a convection-diffusion equation, K is given by $\kappa + \frac{\alpha ul}{2}$, where κ is a positive diffusion coefficient.

Let $N(x)$ denote the Galerkin shape function. In section 3, it can be seen that using the FIC formulation (for space stabilization), the following new shape function is obtained:

$$W(x) = N(x) + \frac{1}{2}\left(\alpha_s lu + \alpha_t l\frac{d\phi}{dx}\right)\frac{dN}{dx} \tag{42}$$

where l is the length of the elements. Setting $\alpha_t = 0$, $\alpha_s = \alpha$ and writing $\frac{d()}{dx}$ as $()'$ yields to:

$$W(x) = N(x) + \frac{\alpha ul}{2}N'(x) \tag{43}$$

If a uniform finite element mesh is defined by: $0 = x_0 < x_1 < ... < x_{N_{el}} = L$, aplying the FIC method in space to equations (39)-(41) will lead to an initial-value problem of the form:

$$M\dot{\phi} + P\phi = 0, \ \forall t \in I\!R_+ \tag{44}$$

$$\phi(0) = \phi^0 \tag{45}$$

where $\dot{()}$ denotes time differentiation.
If now forward Euler is used to integrate equation (44) in time, ϕ^{n+1} is found (assuming ϕ^n is known) by:

$$\phi^{n+1} = \phi^n - \Delta t M^{-1}P\phi^n \tag{46}$$

The stability condition for this equation is well-known and can be found in many references (see for example [5]), but our interest here will be in the analysis of the scheme obtained by taking into account the time stabilization term $-\frac{\lambda}{2}\frac{\partial^2\phi}{\partial t^2}$ introduced by the FIC method. we will then look at the following equation:

$$-\frac{\lambda}{2}\ddot{\phi} + \dot{\phi} + u\phi' - K\phi'' = 0 \tag{47}$$

5.2 Stability analysis of the stabilized equation

Discretizing equation (47) using the FTCS (in the finite element context, using the Galerkin method for space discretization is equivalent, in terms of linear system of algebraic equations obtained, to the centred spatial discretization of the finite difference and finite volume methods) yields:

$$\begin{aligned}-\frac{\lambda}{2}\left(\frac{\phi_m^{n+1} - 2\phi_m^n + \phi_m^{n-1}}{\Delta t^2}\right) + \frac{\phi_m^{n+1} - \phi_m^n}{\Delta t} + u\frac{\phi_{m+1}^n - \phi_{m-1}^n}{2l} - \\ K\left(\frac{\phi_{m+1}^n - 2\phi_m^n + \phi_{m-1}^n}{l^2}\right) = 0\end{aligned} \tag{48}$$

Setting $\lambda = -\delta\Delta t$ gives:

$$\begin{aligned}\delta(\phi_m^{n+1} - 2\phi_m^n + \phi_m^{n-1}) + 2(\phi_m^{n+1} - \phi_m^n) + \frac{u\Delta t}{l}(\phi_{m+1}^n - \phi_{m-1}^n) - \\ \left(\frac{2\kappa\Delta t}{l^2} + \frac{\alpha ul\Delta t}{l^2}\right)(\phi_{m+1}^n - 2\phi_m^n + \phi_{m-1}^n) = 0\end{aligned} \tag{49}$$

Set $c = \frac{u\Delta t}{l}$ (Courant number for $u \geq 0$) and $\gamma = \frac{ul}{2\kappa}$ (grid Péclet number $u \geq 0$) one gets:

$$\begin{aligned}(\delta + 2)\phi_m^{n+1} - (2 + 2\delta)\phi_m^n + \delta\phi_m^{n-1} + c(\phi_{m+1}^n - \phi_{m-1}^n) - \\ \left(\frac{c}{\gamma} + \alpha c\right)(\phi_{m+1}^n - 2\phi_m^n + \phi_{m-1}^n) = 0\end{aligned} \tag{50}$$

ie.

$$\begin{aligned}\phi_m^{n+1} = \frac{1}{2+\delta}\Big\{(2 + 2\delta)\phi_m^n - \delta\phi_m^{n-1} - c(\phi_{m+1}^n - \phi_{m-1}^n) + \\ \left(\frac{c}{\gamma} + \alpha c\right)(\phi_{m+1}^n - 2\phi_m^n + \phi_{m-1}^n)\Big\}\end{aligned} \tag{51}$$

In the von Neumann method, it is assumed that the analytical solution of equations (39)-(41) can be expanded in a Fourier series of the form:

$$\hat{\phi}(x,t) = ae^{-(\xi + i\omega)t}e^{-ikx} \tag{52}$$

where k is the wave number $(= \frac{2\pi}{\lambda_0}$, λ_0 is the wavelength), $\xi = \kappa k^2$ expresses the damping, $\omega = ku$ is the frequency and $i = \sqrt{-1}$.
The discrete counterpart of equation (52) is:

$$\hat{\phi}_m^n = \hat{\phi}(x_m, t_n) = \hat{\phi}(ml, n\Delta t) = ae^{-(\xi^h + i\omega^h)n\Delta t}e^{-ikml} = A^n e^{-ikml} \qquad (53)$$

where ξ^h and ω^h are the algorithmic damping and frequency.
The von Neumann method consists in analyzing the Fourier modes $\hat{\phi}_m^n = A^n e^{-ikml}$ for appropriate wave numbers k. Substituting these modes into equation (51) gives an expression for the (complex) amplification factor $A = \frac{\phi_m^{n+1}}{\phi_m^n}$ as a function of the phase angle $\theta = kl$:

$$A = \frac{1}{2+\delta}\left\{(2+2\delta) - \delta e^{(\xi^h + i\omega^h)\Delta t} + 2ci\sin\theta - 2\left(\frac{c}{\gamma} + \alpha c\right)(\cos\theta - 1)\right\} \qquad (54)$$

We will not use the modified stability criterion defined by Morton [12], ie $|A| \leq 1$ but the original condition $|A| \leq 1 + O(\Delta t)$. We can then write:

$$\frac{1}{2+\delta}\left\{(2+2\delta) - \delta + 2ci\sin\theta - 2\left(\frac{c}{\gamma} + \alpha c\right)(\cos\theta - 1)\right\} \leq 1 \qquad (55)$$

Set $b := \frac{1}{\gamma} + \alpha$. After some algebraic manipulations, equation (55) is found to be equivalent to:

$$4b^2c^2(\cos\theta - 1)^2 + 4c^2\sin^2\theta + 4bc(2+\delta)(\cos\theta - 1) \leq 0 \qquad (56)$$

Depending on the value of b, c and δ the function defined by the left-hand-side of equation (56) has its maximum either for $\cos\theta = 1$ or for $\cos\theta = -1$.

i) $\cos\theta = 1$ yields

$$0 \leq 0$$

ii) $\cos\theta = -1$ yields

$$16b^2c^2 - 8bc(2+\delta) \leq 0 \qquad (57)$$

A condition on the Courant number c, that will insure stability, can be easily found from equation (57) as it is quadratic:

$$0 \leq c \leq \frac{8b(2+\delta) + \sqrt{64b^2(2+\delta)^2}}{32b^2} \qquad (58)$$

or equivalently:

$$c \leq \frac{1+\delta/2}{b} \qquad (59)$$

as the Courant number c is a positive number.
Remarks:

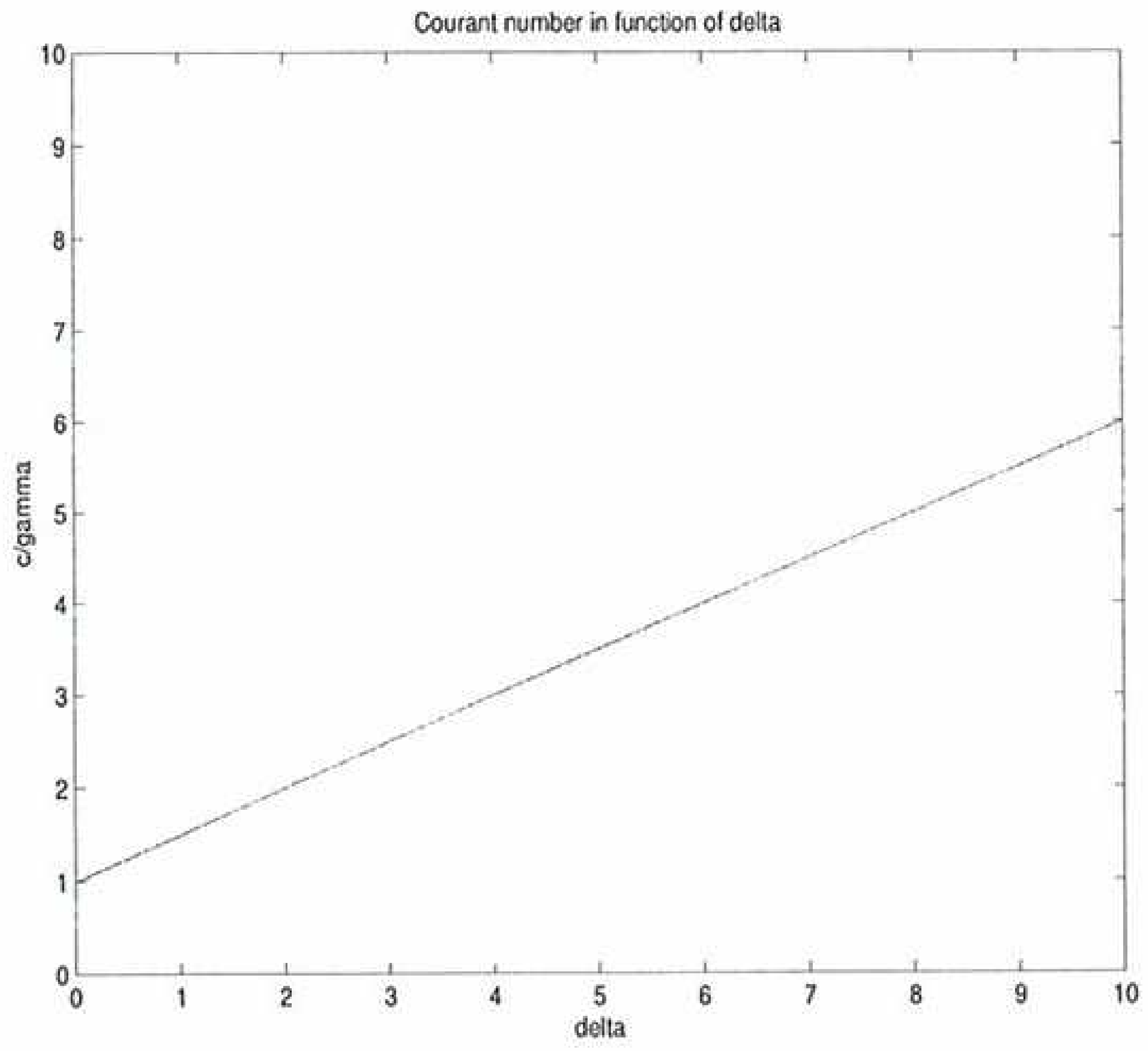


Figure 11: $\dfrac{c}{\gamma}$ as a function of δ

- According to equation (59), the Courant number c is a function increasing with δ, and it can in particular be larger than 1. Figure 11 shows $\dfrac{c}{\gamma}$, or more exactly $c\left(\dfrac{1}{\gamma}+\alpha\right)$ as a function of δ.

- It can for example be noted that for $\delta = 0$, the usual stability condition $c \leq \dfrac{1}{b}$ is obtained.

- For $\delta = 0.5$, $c \leq \dfrac{1.25}{b}$ is obtained.

- For $\delta = 1$, $c \leq \dfrac{1.5}{b}$ is obtained.

Equation (59) can also be used in the opposite way:

$$\delta > 2(bc - 1) \tag{60}$$

Equation (60) can be used in order to have the answer to the following question: given a value of the Courant number c (ie. a time step), what should be the value of the parameter δ in order to obtain a stable scheme?

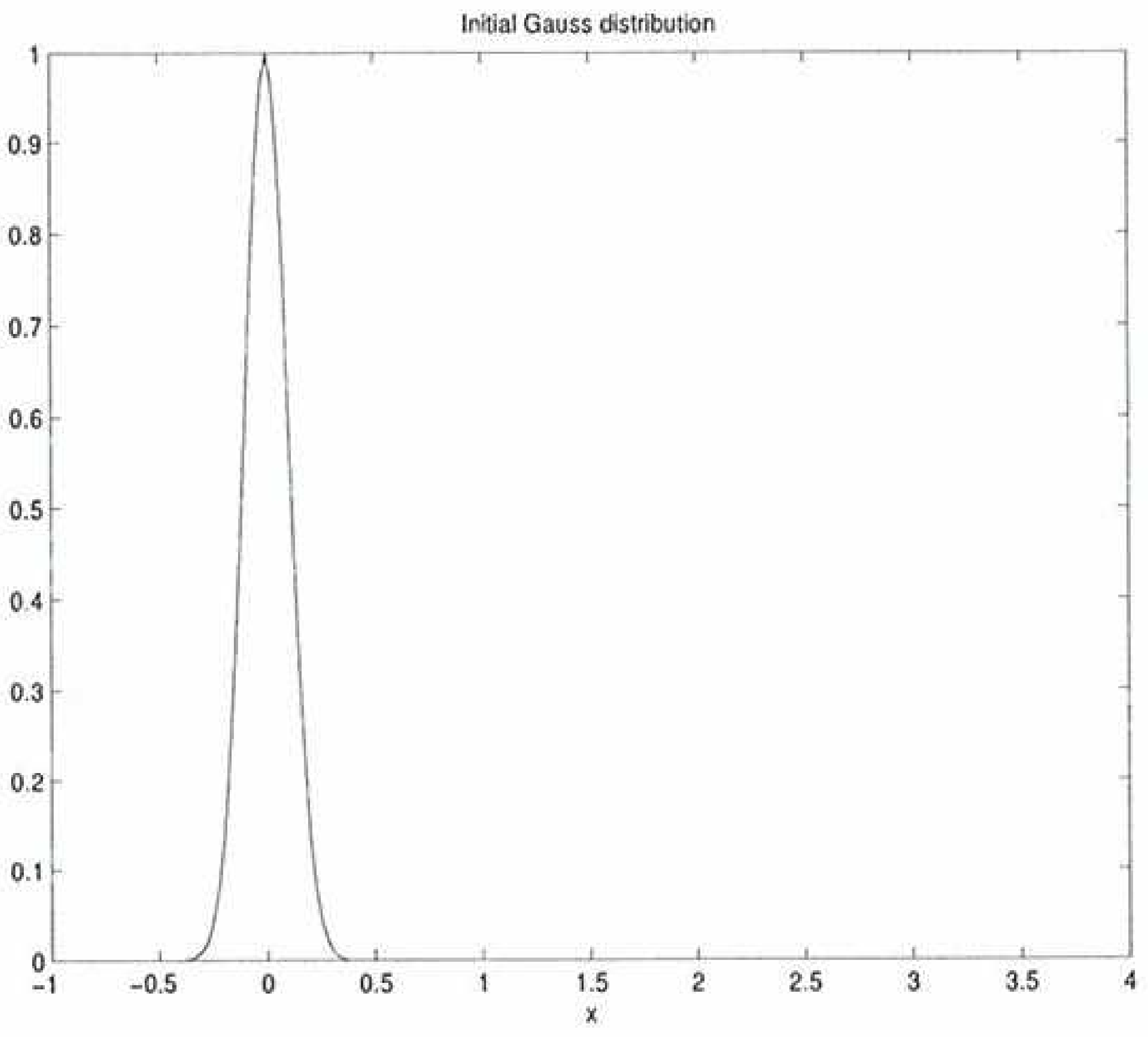


Figure 12: Initial condition.

5.3 Numerical examples

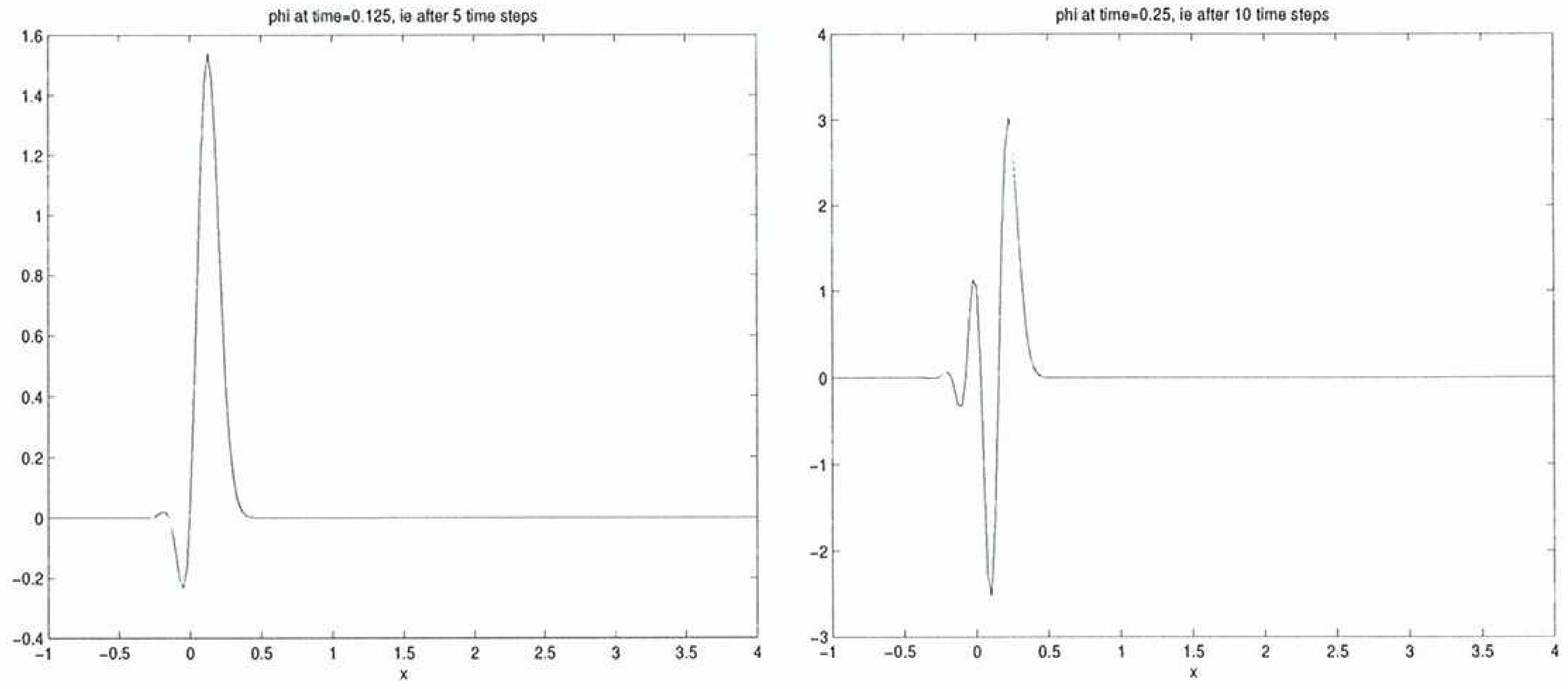
In this section we will consider the pure advection in [0,1] of the Gauss distribution represented figure 12 and given by:

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{61}$$

where the mean μ is set to 0 and the standard deviation σ to 0.1. We use linear elements and set $\alpha = 1$ cf[3]. b is then equal to 1 in the case of the level set equation (which is a pure convection equation).

Figure 13 shows the solution when δ is set to 0. The Courant number is equal to 1. The length of the elements is 0.05, the time step Δt is 0.1 and u is 0.5. The numerical solution is then unstable. Figure 14 shows the computed solution for Courant number $c = 0.9$ and $\delta = 0$. The solution is stable. Some numerical diffusion is present in the scheme since the pic height is reduced as time goes: 1.52% reduction after 10 time steps and 8.51% reduction after 100 time steps. The latter can seem large but the Courant number equal to 0.9 leads to fairly inaccurate results, and in this example, only stability was considered. In figure 15 the computed solution for $\delta = 0.5$ and Courant number $c = 1.2$ is presented. The solution is stable but it is affected by an important amount of numerical diffusion: after 33 time steps, the height of the pic is 0.6657 which means that it has lost approximatively 33% of its height. This tendency is confirmed if the computations is performed further: after 100 time steps the height loss is about 55%.

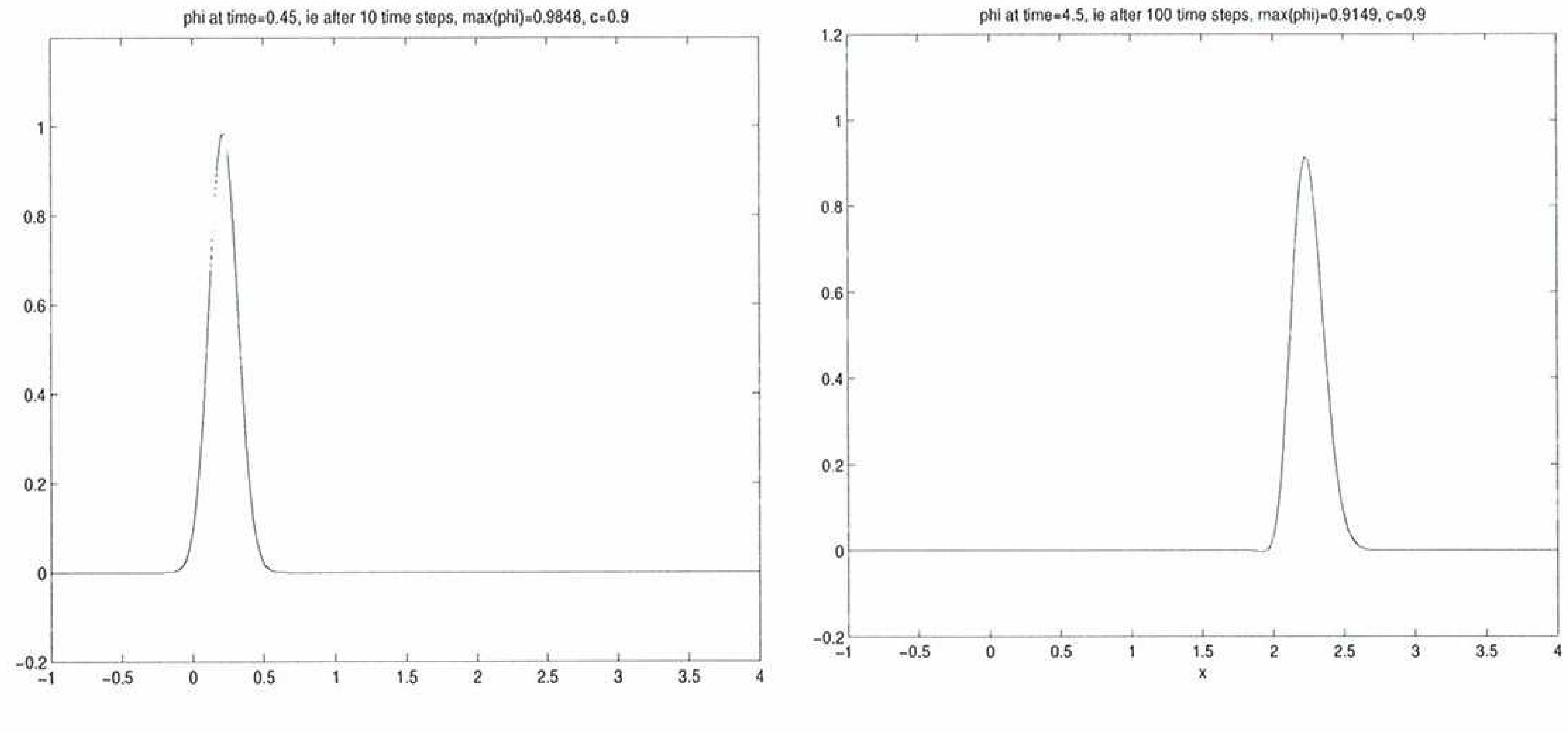
Figure 16 shows the computed solution for $\delta = 0.5$ and Courant number $c = 1.4$. It can be noted that the solution develops instabilities. However these oscillations are not



(a) After 5 time steps

(b) After 10 time steps

Figure 13: Computed solutions for $\delta = 0$, $c = 1$

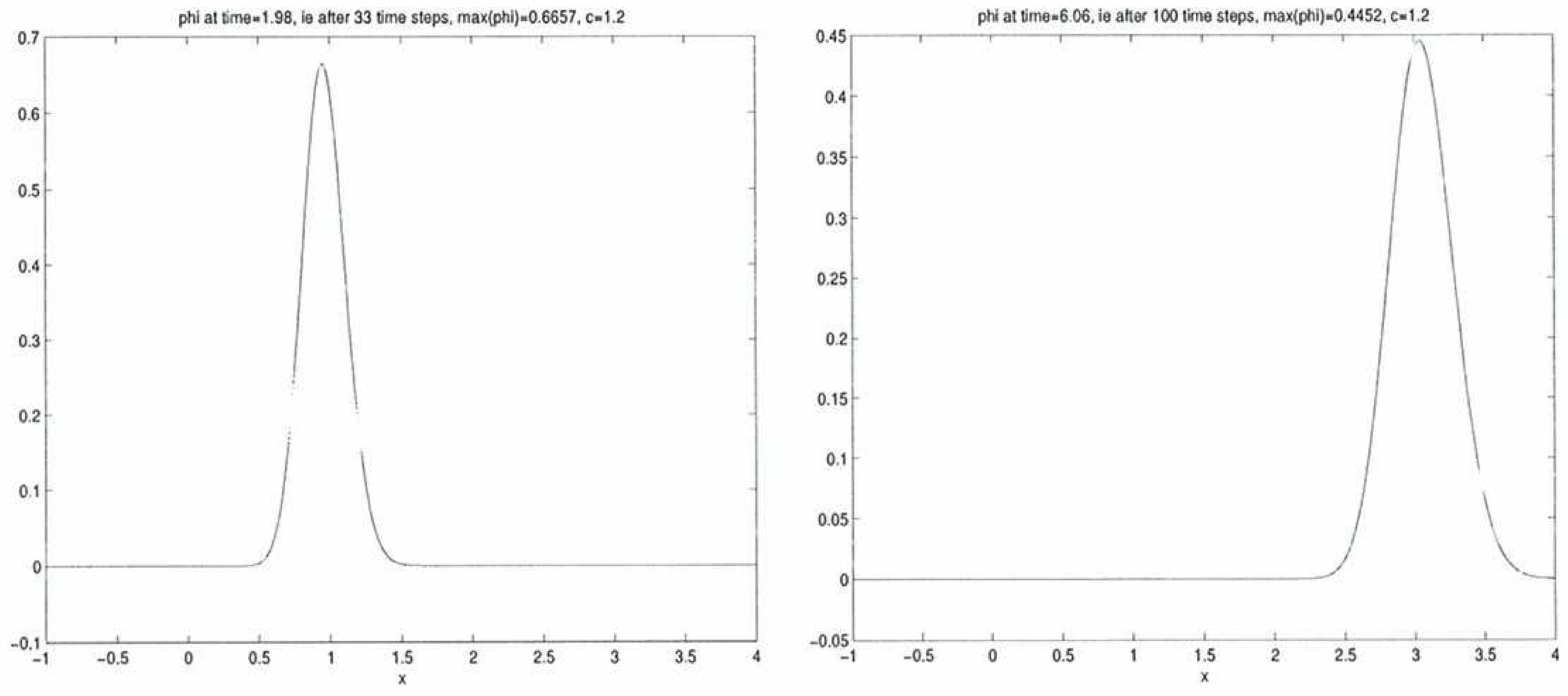


(a) After 10 time steps

(b) After 100 time steps

Figure 14: Computed solutions for $\delta = 0$, $c = 0.9$

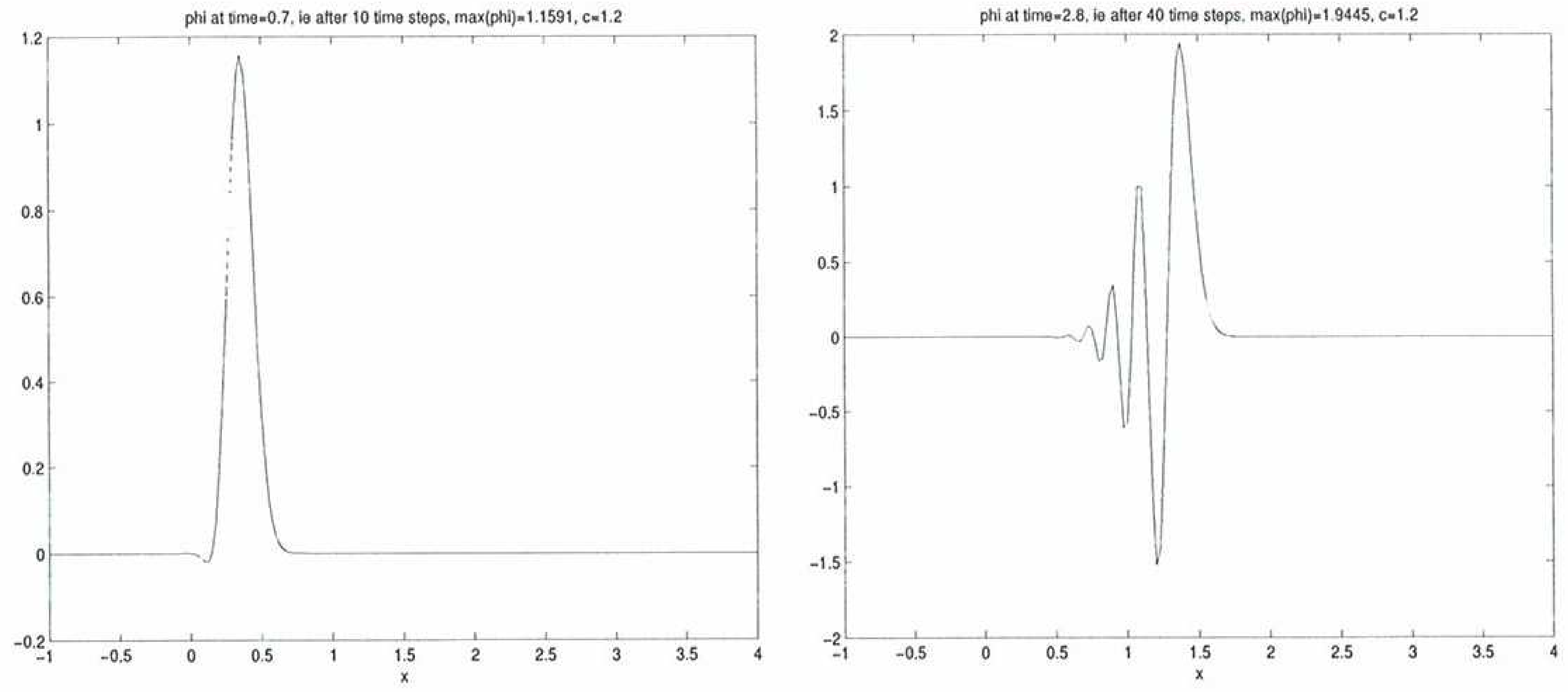
as strong as those observed in figure 13. This means that a certain amount of diffusion is introduced, but not enough: the stabilization parameter δ is not large enough with



(a) After 33 time steps

(b) After 100 time steps

Figure 15: Computed solutions for $\delta = 0.5$, $c = 1.2$



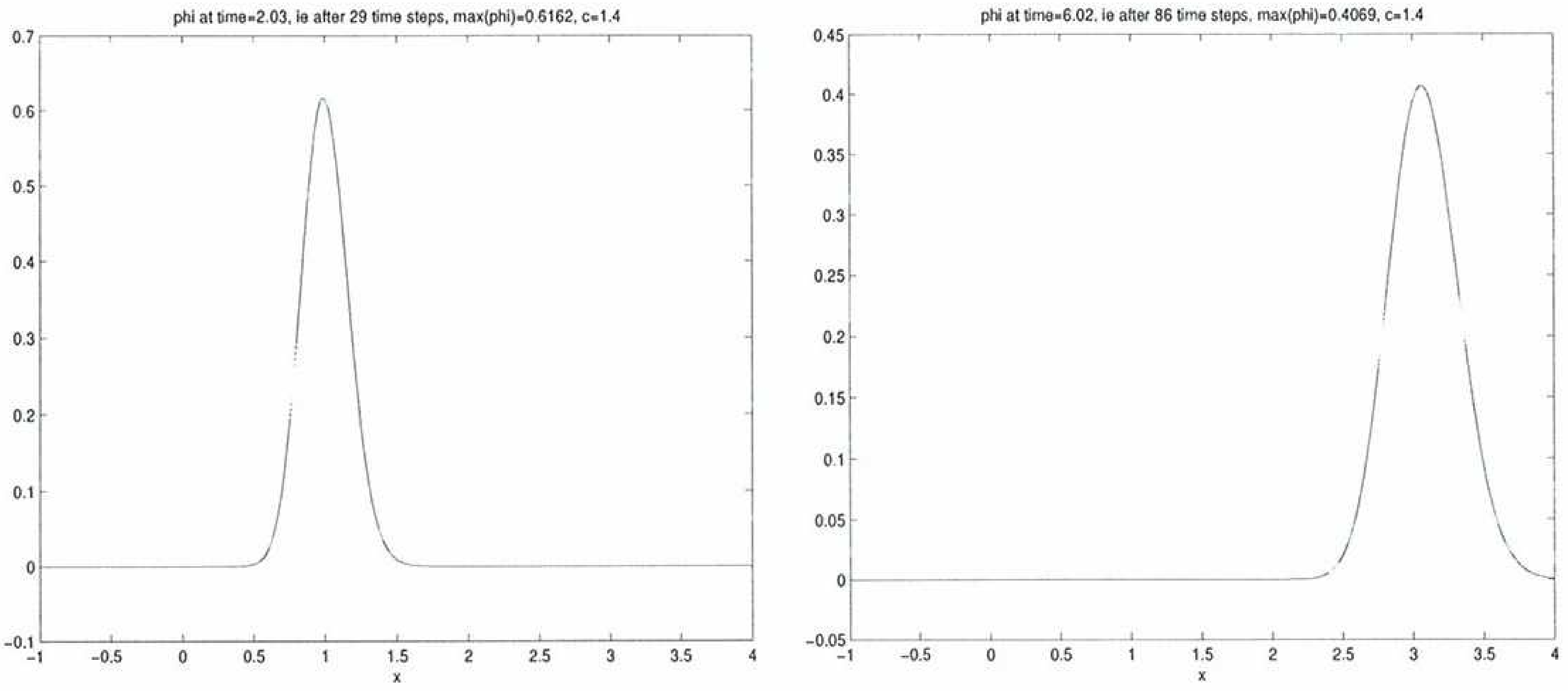
(a) After 10 time steps

(b) After 40 time steps

Figure 16: Computed solutions for $\delta = 0.5$, $c = 1.4$

respect to the value of the Courant number.

Figure 17 shows the computed solution when the stabilization parameter δ is equal to 1



(a) After 29 time steps

(b) After 86 time steps

Figure 17: Computed solutions for $\delta = 1$, $c = 1.4$

and Courant number c is equal to 1.4. We note that the solution is stable as in the case $\delta = 0.5$, $c = 1.2$. We also note that the height loss is increased and is approximatively 38% after 29 time steps and 59% after 86 time steps.

5.4 Multidimensional case

Here, we are interested in the convection-diffusion equation in D dimensions:

$$\frac{\partial \phi}{\partial t} + \sum_{d=1}^{D} u_d \frac{\partial \phi}{\partial x_d} = \sum_{d=1}^{D} \kappa_d \frac{\partial^2 \phi}{\partial x_d^2} \tag{62}$$

on the D-dimensional cube $0 \leq x_d \leq 1$, $t \geq 0$ with $\kappa_d \geq 0$. For problems of practical importance $D = 1, 2$ or 3. However, the results also hold for $D > 3$.
Let us write the FTCS formulation of the equation obtained by applying the FIC method to equation (62):

$$\frac{\lambda_d}{2}\left(\frac{\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}}{\Delta t^2}\right) + \frac{\phi_j^{n+1} - \phi_j^{n-1}}{\Delta t} + \sum_{d=1}^{D} u_d \frac{\Delta_d \phi_j^n}{2\Delta x_d} - \sum_{d=1}^{D} \kappa_d \frac{\delta_d^2 \phi_j^n}{\Delta x_d^2} = 0 \tag{63}$$

where:

- j represents a multi-index $(j_1, j_2, ..., j_D)$,

- Δ_d represents the central first difference operator with respect to the d^{th}-coordinate,
- δ_d^2 represents the central second order difference operator with respect to the d^{th}-coordinate.

Then for each d, we define:

- a Courant number:

$$c_d = \frac{u_d \Delta t}{\Delta x_d} \tag{64}$$

- a grid Péclet number:

$$\gamma_d = \frac{u_d \Delta x_d}{2\kappa_d} \tag{65}$$

- a space stabilization parameter α_d

In addition, we define $b_d = \dfrac{1}{\gamma_d} + \alpha_d$.
A necessary condition for global stability to hold is that the scheme is stable in each direction. Therefore, in light of the analysis performed in section 5.2 for the one-dimensional equation, we write that a necessary condition for stability is:

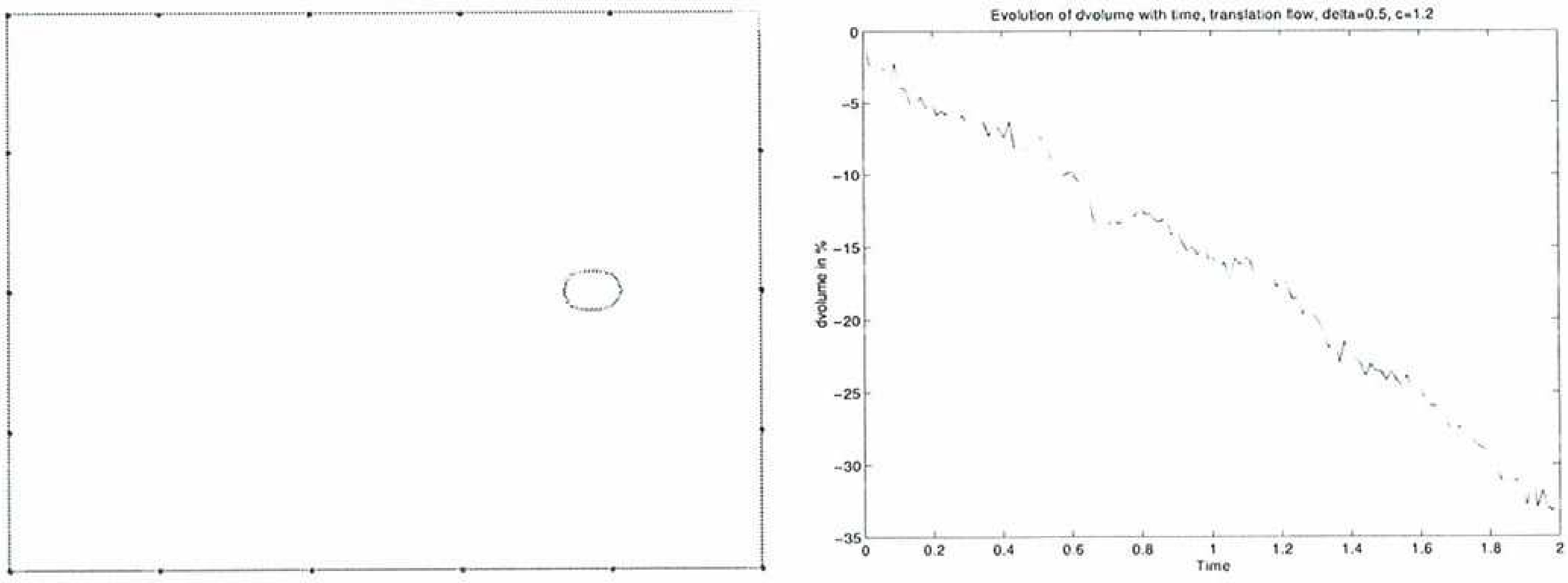
$$c_d \leq \frac{1 + \delta/2}{b_d}, \quad \forall d = 1, 2, 3 \tag{66}$$

We verify equation (66) through numerical examples. We use the same examples as in section 4. In all the examples showed next (which are two-dimensional), the space stabilization parameters are equal to 1, ie: $\alpha_d = \alpha = 1$, for $d = 1, 2$. As in section 4, we will look at the conservation of a circle, for both uniform and circular flow.

5.4.1 Example with an uniform velocity

The fluid velocity is defined as in section 4.1 (ie given by equation (30)). The circle should be advected without changing neither shape nor volume.

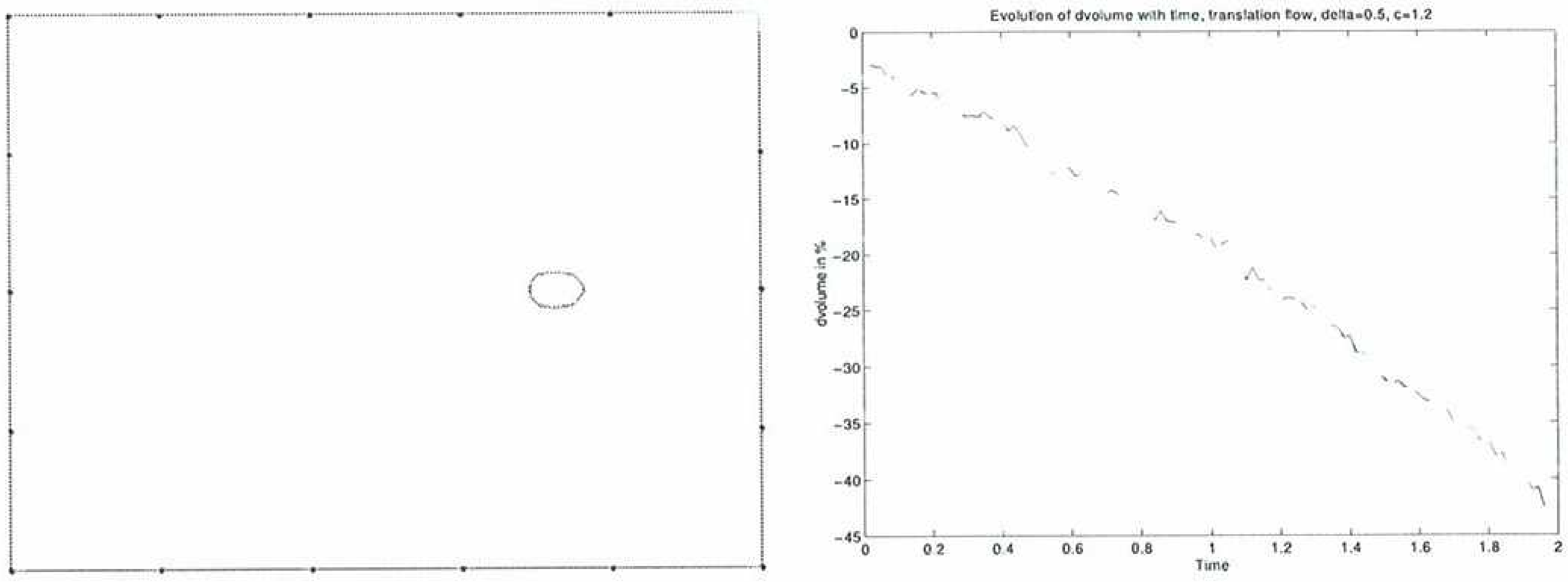
The same features can be noted as in the one-dimensional case: the present method is higly diffusive. In the case where $\delta = 0.5$ and $c = 1.2$, the circle has lost approximatively 30% of its volume after 121 (see figure 18(b)). In the case where $\delta = 1$ and $c = 1.4$, the volume loss is about 38% after 104 (see figure 19(b)). The shape of the circle is also changed, but this is mostly due to the space stabilization. However, most of the change in volume observed here is due to the time stabilization. When only space stabilization is introduced (ie λ set to 0 in equation (10)), the volume loss is a few procent, see for instance figure 7 for the Crank-Nicholson scheme.



(a) Zero level set after at time 1.8 (after 121 time steps)

(b) Evolution of the volume

Figure 18: $\mathbf{u} = \mathbf{i}$. Computed solutions for $\delta = 0.5$ and $c = 1.2$.



(a) Zero level set after at time 1.8025 (after 104 time steps)

(b) Evolution of the volume

Figure 19: $\mathbf{u} = \mathbf{i}$. Computed solutions for $\delta = 1$ and $c = 1.4$.

5.4.2 Example with a circular velocity

The fluid velocity is defined as in section 4.2 (ie given by equation (34)). Here too the circle should be purely advected without changing neither nor volume.
Here the grid is finer than in the case of uniform flow. The reason is that the Reynolds number (based on the circle diameter) is higher here.

Once again the high diffusivity of the method can be observed. In the case where $\delta = 0.5$

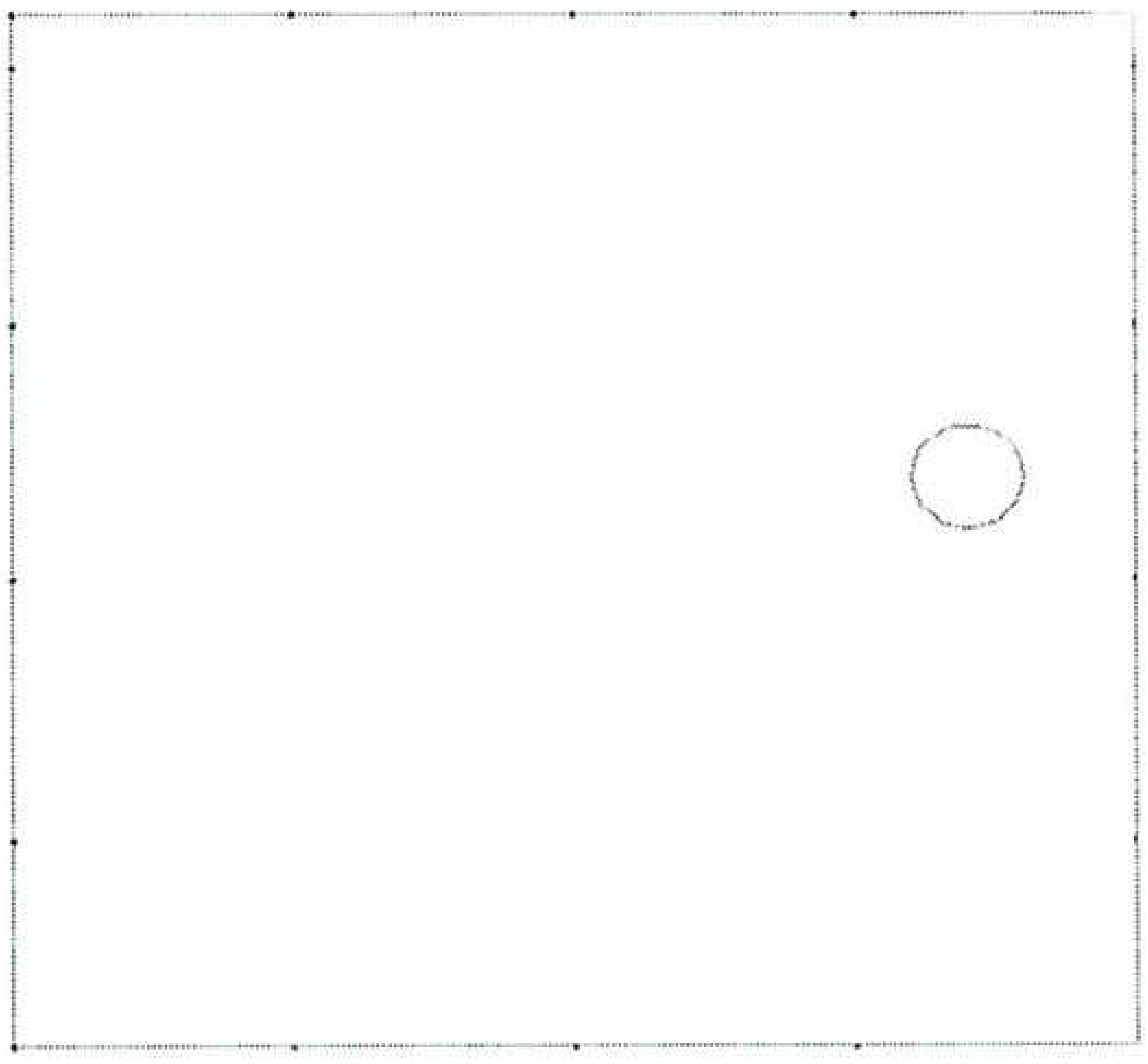
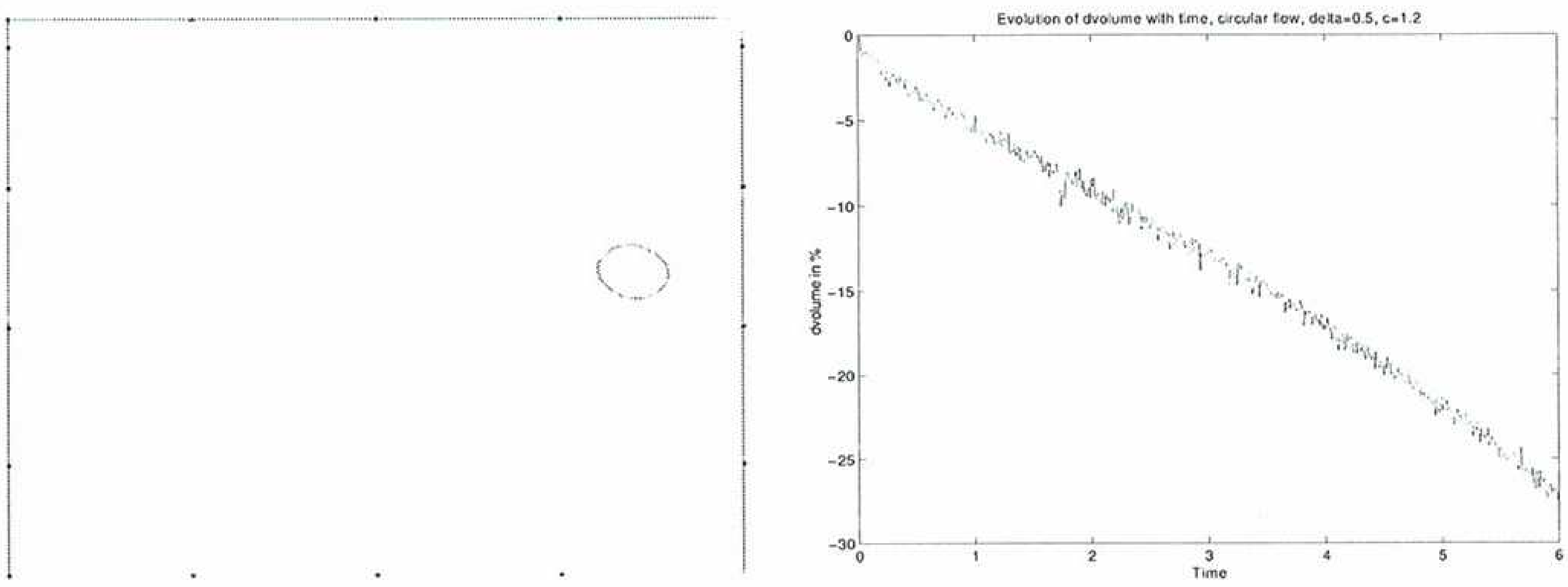


Figure 20: Initial condition for the circular flow.

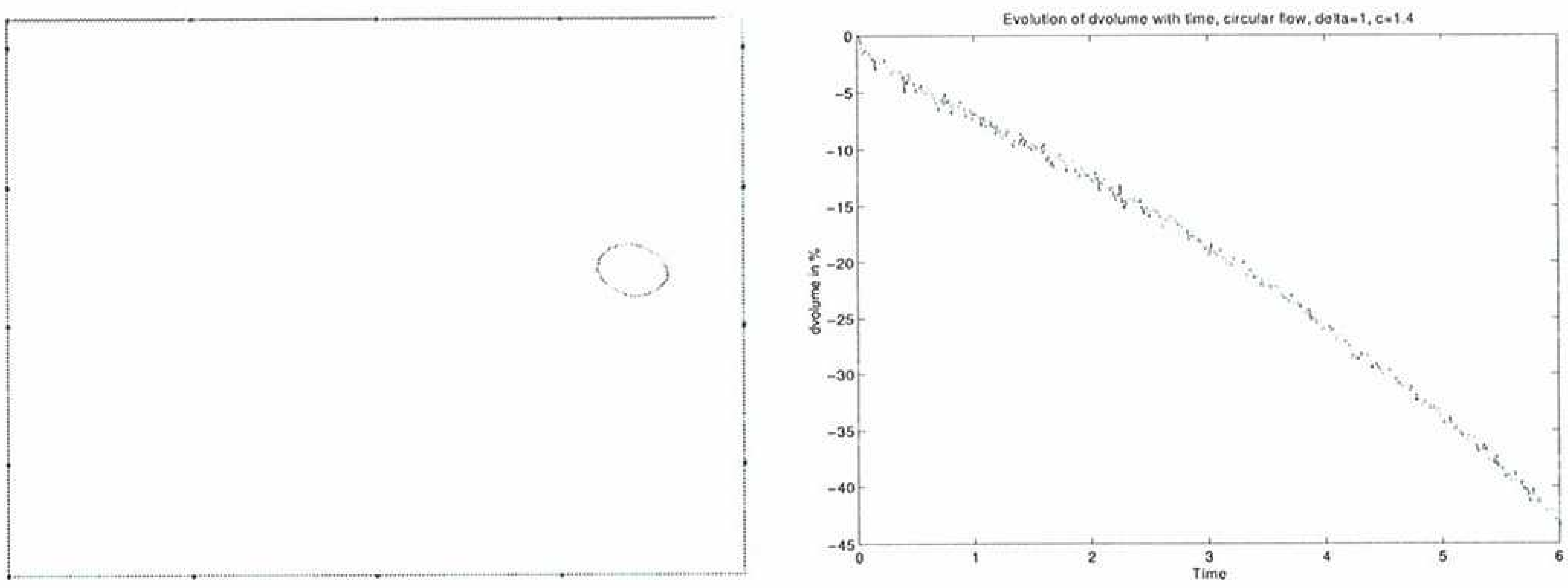
and $c = 1.2$, the circle has lost approximatively 9% of its volume after 1 period ie 162 time steps (see figure 21(b)). In the case where $\delta = 1$ and $c = 1.4$, the loss in volume is approximatively 13% after 1.001 period ie 144 time steps (see figure 22(b)). As the grid is finer than in the uniform flow case, the damping will be less. The difference with figure 10(b) where the mass loss after one period is between 2 and 3% is relatively large.
It can also be noted that the shape of the circle is here slightly more distorted in this case with respect to the uniform flow case, but this is due to the form of the velocity field.



(a) Zero level set after 1 period (after 162 time steps)

(b) Evolution of the volume

Figure 21: $\mathbf{u} = \pi r \mathbf{i}_\theta$. Computed solution for $\delta = 0.5$ and $c = 1.2$.



(a) Zero level set after 1.001 period (after 144 time steps)

(b) Evolution of the volume

Figure 22: $\mathbf{u} = \pi r \mathbf{i}_\theta$. Computed solutions for $\delta = 1$ and $c = 1.4$.

6 Conclusion

Two main issues have been studied here. Firstly, the use of the FIC formulation for space stabilization, and secondly its use to obtain an explicit scheme which is stable for Courant number larger than $\frac{1}{b}$.

When it comes to the first issue, the FIC is shown to be an efficient stabilization method in the case studied (pure convection problem). The stabilisation parameter is chosen in an heuristic manner. However, it can be computed in an adaptive way, and in order to satisfy a given criteria (see [15]).

The other issue addressed is time stabilization of the forward Euler scheme. The use of the FIC formulation allowed us to be able to use the explicit Euler scheme with Courant number large than $\frac{1}{b}$. However, the scheme obtained is not very time accurate. The reason is that an extra numerical diffusion has been introduced in the numerical method and therefore the damping is increased. Moreover, it is a well-known fact that the accuracy of a time integration scheme dicreases with increasing Courant number. Therefore, it seems difficult to use the method as it is when time accuracy is desired. However, the method presented here can be interesting and useful in cases where time accuracy is not important; for example in the case where we are interested in a steady-state solution and that the solution passes through a transient state. The method can then be used in order to be able to "pass" quickly over the transient state, using a large time step.

References

[1] F. Bet, N. Stuntz, D. Hanel and S. Sharma. *Numerical Simulation of Ship Flow in Restricted Water.* Proc. Seventh International Conference on Numerical Ship Hydrodynamics, Nantes, 1999

[2] A.N. Brooks and T.J.R. Hughes *Streamline Upwind/Petrov-Galerkin formulations for convective dominated flows with particular emphasis on the incompressible Navier-Stokes equations.* Comput. Meth. in Appl. Mech. Engrg, vol. 32, pp. 199-259, 1982

[3] Ramon Codina. *A Finite Element Formulation for the Numerical Solution of the Convection-Diffusion Equation.* Monograph CIMNE N^o14, January 1993

[4] A. Cura Hochabaum and C. Schumann. *Free Surface Vsicous Flow around ship models.* Proc. 7^{th} International Conference on Numerical Ship Hydrodynamic, Nantes, 1999

[5] G. Cuvelier, A. Segal and A. van Steenhoven. *Finite element methods and Navier-Stokes equations.* Reidel, 1986

[6] A.C. Hindmarsh, P.M. Gresho and D.F. Griffiths. *The stability of explicit Euler Time-integration for certain finite difference approximations of the multi-dimensional advection-diffusion equation.* Int. Jour. Numer. Meth. Fluids, vol. 4, pp. 853-897, 1984

[7] T.J.R Hughes, W.K. Liu and A. Brooks. *Finite Elements Analysis of Incompressible viscous Flows by the Penalty Function Formulation.* J. Comp. Phys., vol. 30, pp. 1-60, 1979

[8] T.J.R Hughes, L.P. Franca and M. Balestra. *A new finite element formulation for computational fluid dynamic: V. Circumventing the Babuška-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accomodating equal-order interpolations.* Comp. Meth. Appl. Mech. Eng., vol. 59, pp.85-99, 1986

[9] H. P. Langtangen. *Computational Partial Differential Equations - Numerical Methods and Diffpack Programming.* Lecture Notes in Computational Science and Engineering vol. 2, Springer, 1991 Gothenburg, Sweden

[10] R.J. LeVeque. *Numerical Methods for Conservation Laws.* Lectures in Mathematics, Birkhäuser, 1992

[11] R. Löhner, C. Yang and E. Oñate. *Viscous Free Surface Hydrodynamics using unstructured grids.* Proc. 22^{nd} Symp. Naval Hydro., pp. 476-490, Wash. D.C., 1998

[12] K.W. Morton. *Stability of finite difference approximations to a diffusion-convection equation.* Int. J. Num. Meth. Eng., vol. 15, pp.677-683, 1980

[13] A.R. Mitchell and D.F. Griffiths. *The finite difference method in partial differential equations.* John Wiley, 1980

[14] E. Oñate and M. Manzán. *A general procedure for deriving stabilized space-time finite element methods for advective-diffusive problems.* Int. J. Numer. Meth. Fluids, vol. 31, pp.203-221, 1999

[15] E. Oñate and M. Manzán. *Stabilization Techniques for Finite Element Analysis of Convection-Diffusion Problems.* Publication CIMNE no.183, February 2000.

[16] E. Oñate and J. García. *A finite element method for fluid-structure interaction with surface waves using a finite calculus formulation.* Comput. Methods Appl. Mech. Engrg, vol. 191, pp. 635-660, 2001

[17] S. Osher and J.A. Sethian. *Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations.* Journal of Computational Physics, vol. 79, pp. 12-49, 1988

[18] Samuel R. Ransau. *Solution Methods for Incompressible Viscous Free Surface Flows: A Literature Review.* Preprint Numerics no. 3/2002, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway, 2002

[19] J. A. Sethian. *Level Set Methods and Fast Marching Methods.* Cambridge University Press, 1999

[20] M. Sussman, P. Smereka and S. Osher. *A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow.* Journal of Computational Physics, vol. 114, pp. 146-159, 1994

[21] M. Sussman, E. Fatemi, P. Smereka and S. Osher. *An Improved Level Set Method for Incompressible Two-Phase Flows.* Computers and Fluid, vol. 27, no. 5, pp. 663-680, 1998

[22] M. Sussman and E. Fatemi. *An efficient, interface preserving level set re-distancing algorithm and its application to interfacial incompressible fluid flow.* SIAM J. Sci. Comput., vol.20, number 4, pp.1165-1191, 1999

[23] M. Sussman and D. Dommermuth. *The Numerical Simulation of Ship Waves Using Cartesian Grid Methods.* Proc. 23^{rd} Symp. Naval Hydro., Val de Rueil, France, 2000

[24] T.E. Tezduayr & D.K. Ganjoo. *Petrov-Galerkin formulations with weighting functions dependent upon spatial and temporal discretisation: application to transient convection-diffusion problems.* Comput. Meths. Appl. Mech. Engrg., vol. 59, pp. 49-71, 1986

[25] M. Vogt and L. Larsson. *Level Set Methods for Predicting Viscous Free Surface Flows.* Proc. Seventh International Conference on Numerical Ship Hydrodynamics, Nantes, 1999

[26] O.C. Zienkiewicz and R. Taylor. *The Finite Element Method.* McGraw-Hill, London, 1989